A Secure e-Prescription System Based on NFC

by

© *MohammadKazem Poozesh*

A thesis submitted to the

School of Graduate Studies

in partial fulfilment of the

requirements for the degree of

Master of *Science*

Department of *Computer Science*

Memorial University of Newfoundland

November 2016

St. John's Newfoundland

Abstract

The main purpose of this thesis is to introduce a secure e-prescription system which is able to protect patients and their physicians, who have privacy concerns in using paper-based prescription. Patients have concerns with disclosing their health information that might be a source of embarrassment, lack of confidence or damage to the individual's reputation. On the other hand, drug companies want to figure out physicians prescribing patterns to persuade them for prescribing their drugs by sending their representatives to them. This thesis introduces a new e-prescription system based on Near Field Communication (NFC) technology, using smartphone and cryptographic tools such as digital signature to protect the identity of both the patients and physicians. The proposed system is implemented as a mobile application to use on smartphones and other hand-held devices. In order to achieve this goal, this thesis adopts a digital proxy signature and a short group signature. The proxy signature helps the patient to delegate their signing right to another party in order to collect their prescription when they are not able to do so. The short group signature helps the physician to sign the prescription anonymously in a group of physicians, which will protect the identity of the physician. The short group signature has been used in the implementation. This signature, which is shorter than RSA-based signatures, helps to suit better this application on handheld devices, which have power and computation constraints. The implementation of this system also uses NFC on smartphones, which will be used at pharmacies by the patients as the proof of their identities. As a result, without revealing the real identity, patients will be able to

collect their prescriptions.

Acknowledgements

First, I would like to thank my supervisor, Prof. Saeed Samet for the continuous support of my MSc study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing my thesis. I would also like to thank my family for supporting me spiritually throughout my graduate study. Finally, I must express my gratitude to all my PhD and MSc friends at the e-Health Research Unit for providing me academic assistance and reviewing my thesis. My thanks also go to one of my friends, Mark, for proof reading my thesis.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

The digital era has changed every aspect of our life very quickly. Nowadays, people prefer digital systems because of their higher security and performance. Some examples of digital systems currently widely in use are banking systems and navigation systems. In medicine, some of the first generation of these systems include Electronic Health Record(EHR) and the Electronic Medical Record(EMR) which refer to digital records of patient's information like medical history, allergies and laboratory results. After successful experience of EMR and EHR, which improved the performance of the healthcare, a lot of efforts have been made to move the infrastructures from traditional prescription systems to digital-based to benefit fully from their security and higher performance. According to [4] , in the US, 1.07 million professionals are using e-prescribing systems and it is growing. Managing prescriptions digitally is more convenient and more secure than the traditional paper-based prescription systems.

For example, working with smartphones helps patients to access medical information at all times even while traveling. E-prescription systems are usually integrated with EHR and EMR. This helps physicians to check the history of a patient's medical record to prescribe more accurately. E-prescription systems also can be implemented independently. From the privacy point of view, it is worthwhile to consider a secure system to preserve the privacy of both the patient and physician. It prevents non-authorized people from accessing patients' and physicians' confidential, sensitive information.

1.2 What Is E-prescription?

e-Prescribing systems are network-based systems replacing the traditional paper-based prescription systems. These systems allows physicians to generate and transmit accurate and legible prescriptions to pharmacies electronically. It can also be used to renew electronic prescriptions. These systems must be secure for preserving the privacy of both patients and physicians. For this purpose, the proposed system uses cryptographic tools such as digital signatures and cryptographic protocols like pairing based cryptography. These systems are growing very fast because of their improvement of healthcare quality. According to [25], the European Union(EU) is going to implement a cross-border healthcare system in which all EU citizens will be able to access e-Prescription systems anywhere in Europe. In the United States, according to Surescript, which owns the largest company that is active in the e-Prescribing area, nearly 317,000 office-based physicians used e-Prescribing in 2012 [4].

1.3 Benefits of E-prescription Systems

- Enhance the quality of healthcare:

  Paper-based prescription can have negative effects on the quality of healthcare. Handwritten errors are very common in paper prescriptions, which can result in delivering wrong medication to patients. On the other hand, the misreading of prescriptions is another problem associated with it at pharmacies. No access to drug reference information has negative effects on the decision of physicians as well. E-prescription systems can provide a method for managing the medication process. In this process, the system checks the patient's current medication status in order to find any drug-drug reaction, drug allergy, body weight, age, drug appropriateness, dosage correctness, adverse reactions etc.

- Reduce phone calls and call-back to pharmacies:

  Even after prescribing medications, physician's offices receive callbacks from pharmacies with clarification and refill requests. These callbacks can waste huge time of staff responding to phone calls.

- Refill requests can be done very quickly through the e-prescription system. The system can automatically receive refill requests, check them, and send results back to pharmacies instantly.

1.3.1 Smartphone Aspects

Smartphones are changing our lives and have become an important part of our daily life. We use smartphones for many aspects, from making a call to check for our online banking accounts. Even 20 years ago, some of these functionalities were not even possible with other devices, but now we have all of these functions in one device. This study is using smartphones because of mainly three reasons. First, these devices are very secure in compare to paper-based prescriptions and smart cards. Some new models are using biometrics features like fingerprints and image processing in order to log into the device. Second, they have some features including alarms and notifications that help the patient to be aware of her medications at all times. Final usage of the smartphone is at the pharmacy; the patient uses her smartphone to prove her identity via Near Field Communication (NFC) technology. This technology helps to transmit data securely to the pharmacy without revealing the real identity of the patient. This assures that the right person receives the prescription. Chapter 3 discusses NFC technology, which is now available on new smartphone models. The other advantage of using smartphones is that the patient will be able to track the history of her drugs easily.

1.4 Motivation

Despite the importance and huge share of investment in the healthcare , there still exist some noticeable shortcomings. In this study, first, we introduce these problems then it presents a secure e-prescribing system aiming to generate and transfer prescriptions digitally from health providers to pharmacies while reduce the number of occurring problem in the traditional way. The following two subsections demonstrate the existing shortcomings in this area, which addressing them was the encouragement of accomplishing this thesis.

1.4.1 Preventable Medical Error

According to the National Academy of Science, there are 400,000 deaths annually due to various preventable medical errors [24]. According to [29] , medical errors are the third main cause of death in the US after heart disease and cancer. According to [29], preventable medical errors kill more people annually than AIDS, breast cancer and drug overdoses. Consequences of these errors might range from an immediate death or accelerate an individual death.

Drug prescription in healthcare area can be transformed with the help of digital and information technology to prevent these type of errors. In the paper-based prescribing system, the common errors are including wrong drug, wrong dosage and not checking allergies [26]. In a study, there were four prescription errors with adverse effects out of 1000 prescriptions in a teaching hospital [37].

1.4.2 Preserving Privacy

The other issue with the traditional means of drug prescription is security and privacy concerns. Information about a patient's health is highly sensitive. Patient privacy in paper-based prescription might not be properly preserved. For example, when a patient brings a prescription to a pharmacy, it is not hard to extract information on her health situation based on the prescribed. For example, if the patient orders prescriptions for Efavirenz and Tenofovir Emtricitabine, the pharmacist will find out that the patient is dealing with HIV treatment [32]. In another example, if a patient receives a prescription for Olanzapine(Zyprexa), she is under treatment for a mental illness. [32]. Preserving the privacy of the physicians is another concern that should be taken into account. Pharmaceutical companies tend to extract physicians' information from prescriptions to increase their revenue. These companies use the extracted information from prescriptions to identify physicians who are likely to prescribe their drugs to target more accurately their marketing efforts. They use this information to send their representatives to those physicians who might be interested in their drugs [37]. Suppose a company releases a new Triptans drug for treating migraines, some physicians tend to prescribe a pain reliever for migraines; some physicians prefer to prescribe Triptans for migraine treatment. Companies want to know which physicians prescribe Triptans and who prefer to use existing migraines drug [32]. They also want to know which physicians prescribe new drugs to send their representatives to the right physicians upon releasing a new drug [32]. Sales representatives constant contacts with the physicians might have a negative effects on the prescribing process, because they might persuade physicians with incentives to prescribe their products.

This thesis proposes a new e-prescription system based on NFC, which is a wireless contact technology introduced by [38]. This technology is used to proof the identity of the patient. At pharmacies, the patient no longer needs to show any ID, and just needs to bring her smartphone close to a detector to confirm their identity. Some advantages of using NFC technology in e-prescribing systems include:

Ease of access: When using NFC technology, there is no need to carry several cards or dig through a wallet to find the right smart-card.

Security: Using smartphones is more secure than other items like smart-cards. In a case of smartphones lost, strangers cannot access and illegally use their stored data. Some smartphones need biometric features like fingerprints or the owner's picture for logging in, which make it very hard to log into it.

1.5 Contents Outline

The remainder of the thesis is organized as follows: Chapter 2 reviews work previously done in e-prescription area and compare them to this proposed system. Chapter 3 discusses the details of the cryptographic tools used in the implementation, such as digital signatures. Chapter 4 demonstrates structure and architecture of the proposed model. Chapter 5 explains implementation details and methods. In Chapter 6, experimental results will be showed. Finally, Chapter 7 is the conclusion and future work.

Chapter 2

Background and Related Work

This chapter, first specifies privacy concerns in health area and explain the addressing methods offered by two legal acts in a non-technical view. This review is neccessary because in the proposed model there exist some non-trusted parities which legal acts restrict them mainly from disclosing individual's information. In the subsection 2.2 demonstrates two standards related to e-prescribing systems which every system of this type should be comply with them. In the end, two major work related to e-prescribing systems will be reviewed.

2.1 Health Privacy Concern and Addressing Methods

In a very general form, privacy means the right to be alone. Complexity and difficulty of life associated with advancing in societies forced human being to have a private environment to think about this rapid changes [39].Nowadays number of companies have been interfering with private life of people and making money by breaking their privacy. For instance, newspapers and magazines try to publish the private matter of celebrates to make huge profits. Health area is one of these private environment which contains very sensitive and personal information. Patients are not inclined to disclose their information to the public and non-reliable companies. The privacy concerns over how health organizations access and manage sensitive Personal Health Information(PHI) have been source of concern for decades. Privacy over health data for people is much more important than other areas, because revealing these personal data might lead to harmful effects on people on daily basis life such as: embarrassment and lack of confidence. In traditional managing of health data, patients do not have full control over their information, because their health information are kept in paper-based format in some semi-trusted parties' information centers. Since health organizations store this information in a plain and non-secure way, people that have access to this information are able to read and reveal this information.

With the advent of the digital era, areas like banking, shopping and navigation transformed enormously in terms of accessibility, simplicity and security. Healthcare sector have benefited from the digital area as well. For example, managing and storing of health data in EMR and EHR systems is growing among healthcare organizations. However, this advancement in the electronic healthcare should not be at cost of patients' privacy.

2.1.1 Personal Health Information Protection Act

The personal Health Information Protection Act (PHIPA) of Ontario in 2004 forces health organizations to protect personal health information in their data centers in order to ensure this information is stored, transferred and disposed in a secure way.

Steps that should be taken into account consist of physical security, technological, and administrative control [5].

Physical security steps include:

- Locked physical information storage centers like cabinets or server rooms
- Restrict office access and use an alarm system in sensitive places

Technological security means consist of:

- Using authorization like passwords, IDs, pass cards
- Using Encryption
- Firewall and Anti Virus software

Administrative protections consist of:

- A written set of security rules
- Employee training
- Level of access

2.1.2 Ethical Force Program (EFP)

Our purpose in this section is to describe some existing basic and non-technical rules to protect PHI extracted from the Ethical Force Program (EFP). EFP was first introduced by the American medical association to improve health care quality by encouraging involved parties to respect more ethical behaviors. EFP extracted from a more general set of rules called Fair Information Practices(FIP). Many organizations, not just health care providers, benefit from FIP for their data management and data privacy. In 2000, EFP released a set of recommendations to better manage and protect health information as follows: [35, 36].

- De-identification: Before any process on PHI, providers may use de-identification methods to help reduce privacy concerns, although it might still be some privacy concern in these methods. For more details on these methods, readers can refer to [19].
- Transparency: Every party involved with PHI must clearly explains their policies and make their procedures publicly available.
- Consent: In Most of the time, for every operation on PHI, consent of individuals should be obtained by trustees. Otherwise, a formal process should be taken into account in order to waive this consent.
- Collection limitation: Any healthcare provider processing on PHI should limit their collecting to just information they need at the moment. For further use, they should mention it clearly at the time of obtaining consent.
- Security: The Trustee must provide adequate means of security to protect PHI.

For example, they might have some firewall on their electronic systems or use cryptographic tools to encrypt health data for transmitting to other parties.

- Level of access: Every health care provider should not have the same level of access to PHI. For example, a researcher and a nurse should have their own restrictions for viewing the PHI. Access should be kept at the minimum level of need and no unnecessary information should be available for individuals within a healthcare provider.

- Data Quality: Healthcare providers should assure the quality of their data. It should be accurate and up-to-date. To achieve this goal, providers must periodically update their PHI databases.

- Accountability: There must be a mechanism in place to ensure that the healthcare provider obeys all of the above recommendations when dealing with PHI to preserve that privacy of individuals properly.

2.2 Related e-Prescribing Standards

2.2.1 National Council for Prescription Drug Programs (NCPDP)

NCPDP is an ANSI-accredited standards development organization providing forums and standards in healthcare areas, especially in pharmacy business solutions [3]. They provide standards, which are used in pharmacy processes, payer process, and e-Prescribing. Their standards are complying with Health Insurance Portability and Accountability Health Insurance Portability and Accountability Act (HIPAA) and other legal acts. The NCPDP standards include several items that are used in pharmacy service messaging and other healthcare areas:

- Telecommunication Standard: Used for eligibility of the communications between pharmacies, insurers and benefit management companies (PBMs).

- SCRIPT standard: Is used to transmit prescriptions from physicians to pharmacies. Type of this messages include New, Change, Renewal, Cancellation, and Fill Status.

- ASC X12N-2701/271 Standard: Is used for eligibility and benefits of the communications among professionals, dentists, institutions and health plans.

- NCPDP Formulary and Benefit Standard: This standard is about patient benefits information used by physicians.

2.2.2 RxNorm

RxNorm, produced by National Library of Medicine (NLM), provides standard names for clinical drugs and drug related devices [28]. RxNorm enables digital health systems including e-Prescribing systems to communicate efficiently and unambiguously. It helps these systems to communicate regardless of their software and hardware infrastructures.

2.3 Related Work

2.3.1 Computerized Physician Order Entry

Nowadays, the digital transformation changed every aspects of our life ranged from personal to social aspects. Perhaps, in medicine area, the best example of these changes reflects in advent of computerized physician order entry (CPOE). In CPOE systems, physicians can directly place orders electronically, and transmit them to the recipient including a nurse or other healthcare staffs. These systems emerged 10 years ago, before that, most of the orders were handwritten, but now a majority of hospitals are using CPEO [1]. The original purpose of these systems was to prevent medications errors in inpatient and outpatient practices, but modern version of them can have other functionalities like tests and consultation. According to [9], approximately 90 percent of medications errors occur at the time of writing prescriptions by physicians and reading prescriptions by pharmacists. These errors include poor handwriting, ambiguous abbreviation or wrong order details by clinicians. With the advent of CPEO, these errors decreased in all stages. According to [33], these medications errors reduced by 48 percent with use of CPEO in comparison to paper-based systems. Some of the advantages of CPEO are as follows:

- Decreasing handwriting problems
- Reducing similar drug names problem in prescriptions
- Preventing adverse drug-drug reaction
- Faster transmission of prescriptions to the recipient

- Ability to recommend alternative tests or drugs which might be safer for a special case

In spite of advantages of CPEO in decreasing medication errors, there are some issues with these systems too. It is shown in [40] that CPEO itself might cause some errors, for example it can increase workload for clinicians that might lead to the lack of concentration on the main tasks. CPEO systems are used in local network of hospitals connecting physicians and other healthcare staffs in compare to e-prescribing systems which connects patients and physicians directly.

2.3.2 Background of e-prescription Systems

In 2002, Giuseppe Ateniese and Breno de Medeiros [8] introduced an anonymous e-prescription system that adopts a group signature-based model. The group signature model is a cryptographic method introduced by Chaum and Van Heyst [15] that will be discussed in greater detail in Chapter 3.

In 2004, Yang et al. [41] proposed an e-prescription system based on smart cards aimed to preserve the privacy of both patients and physicians. According to their proposed system, after a patient visits the doctor, the doctor connects to the central medical record database to check allergies and possible conflicts between drugs. The physician then sends the prescription to the group manager, who signs the prescription on behalf of the doctor. the digital signature used in this model assures preserving the identity of the physician. The prescription is next forwarded to the pharmacy and added to the medical record of the patient. Then the pharmacist first checks applicable laws and regulations with the insurance company and once authorized by the patient, dispenses the medication. In Yang et al.'s model, the smart card serves two main roles. It acts as a portable data repository that stores personal medical records and insurance information. It is also a signature generating tool that electronically signs the prescription pad when the patient visits the pharmacy. Another important feature introduced in their work is the use of a proxy signature to delegate signing rights to a third party. It is common that patients are unable to collect their prescriptions and it has to be collected by relatives or agents. As Yang et al. pointed out in their system, patients do not need to pass their personal smart card to others. They can just delegate their signing rights to another person. The third person uses their smart card to collect the prescription.

Yang et al. [2004] claims there is no such group signature that can revoke the identity of the doctor, so it is necessary to sign all prescriptions by the group manager. This scenario is shown in their signature's model in Figure 2.1. The prescription should be signed by the group manager in order to protect the identity of the physician. The proposed method for signing all prescriptions by the group manager can result in a bottleneck, affecting its overall performance.

In 2012, Hsu et al. [23] introduced another smart card based e-Prescribing system. They added a role of a chemist to the Yang et al. system for settling medications disputes. They also used an identity-based group signature and ID-based proxy signature. In an ID-based system, introduced first by Adi. shamir [34] , the public key is simply a user identity such as email or phone numbers.

The proposed system in this thesis resolves the issue of bottleneck in Yang's model with help of the short group signature introduced by Dan Boneh et al. [10]. With using the short group signature, the group manager issues a unique key for each

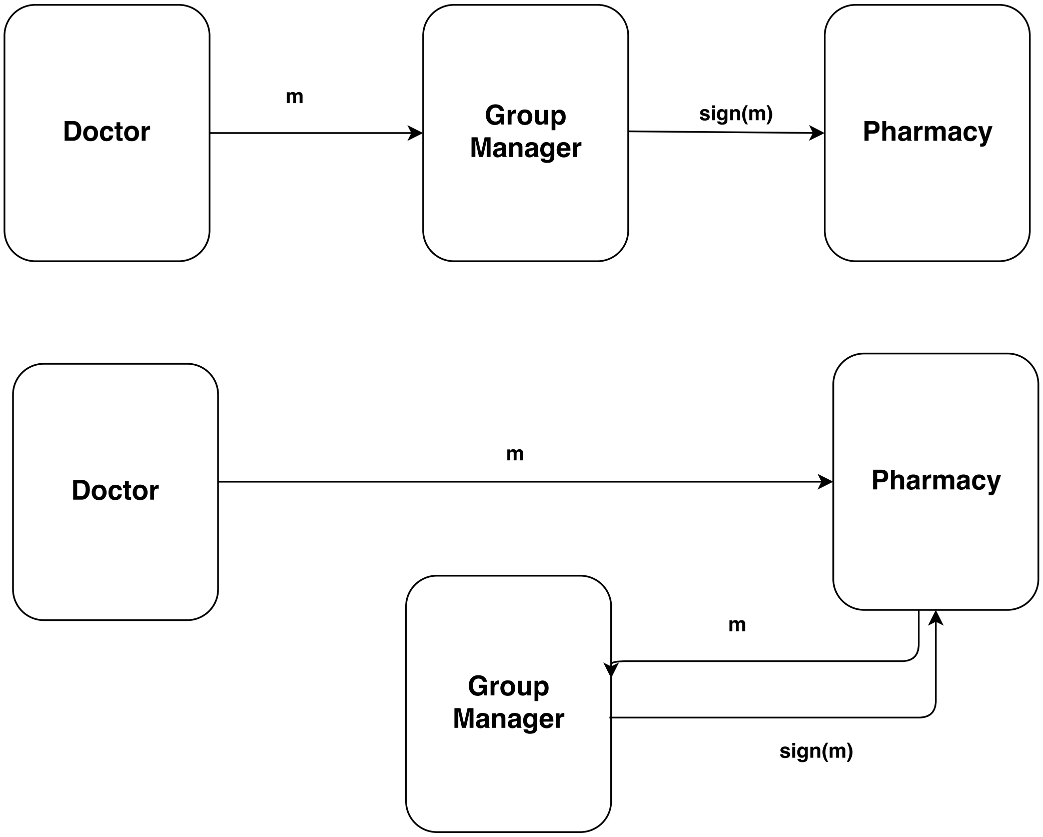


Figure 2.1: Yang et al. model.

doctor. This enables doctors to sign prescriptions by themselves, while still allowing the group manager to open their signature with a master key in dispute situations. Another advantage of this system over Yang et al. is the use of smartphone and NFC technology instead of a smart card. The smart card technology is an old and passive technology with small memory capacity. The retrieval of data on a smart card requires the use of detectors, whereas the smartphone can easily retrieve all prescription data for the patient easily. Patients will be able to trace the history of their medicine in smartphones and receive appropriate notification regarding their medicine.

Chapter 3

Preliminary and Definitions

3.1 Pairing Based Cryptography

This section introduces pairing based cryptography (PBC), which will be used in the implementation part in this thesis. Recently, PBC has been used in many applications specially after introducing identity-based cryptography by Boneh and Franklin [11, 31]. For fully understanding PBC, there are some preliminaries to explain such as cyclic groups, elliptic curves and bilinear pairings as follows.

3.1.1 Cyclic Group

This section introduces cyclic groups because elliptic curves are defined on them. A cyclic group is a set of elements that can be generated by one element called the generator. For example, a group of integer z can be generated by adding 1 to itself repetitively. In this way, the element 14 can be generated by adding 1, 13 times to itself. So, the group of integers z is an instance of additive groups.

3.1.2 Elliptic Curves

Elliptic curves are widely used in cryptography as a basic construction in many cryptographic tools. Elliptic Curves Cryptography (ECC) presents a public key cryptography with relatively smaller key size with equivalent security in compare to RSA-based cryptography. This feature is useful for implementation systems with power constraints and high speed. One of the cryptographic methods that is based on elliptic curves is pairing based cryptography which has been used in this study and will be explained in detail later in this chapter.

Elliptic Curve Definition: If F_p be a finite field with prime order p, an Elliptic curve E over F_P is defined by an equation of the form:

$$y^2 = x^3 + ax + b$$

where $a, b \in Z_p$ **;** $4a^3 + 27b^2 \neq 0$ **mod** p **and**

Z_p **is a finite field of integers with p members**

All points $(x, y) \in (Z_p, Z_p)$ which satisfies the above equation, along with a point at infinity ∞ forms a set of points named $E(Z_P)$.

3.1.3 Bilinear Pairing

Suppose we have two cyclic groups G_1, G_2 with two generators $P, Q \in G_1$ with same order q. A bilinear pairing is a map $e : G_1 \times G_1 \rightarrow G_2$ with three main properties as follows:

- Bilinear: $e(aP, bQ) = e(P, Q)^{ab}$. This is the main property which will be used in verification of a digital signature.

- Non-degenerate: There exist $P, Q \in G_1$ such that $e(P, Q) \neq 1$

- Computable: For deriving $e(P, Q)$ for $P, Q \in G_1$ there is an efficient algorithm

A more complete resource for these definitions can be find in [42].

3.1.3.1 Tate Pairing

There are two main pairings named Weil and Tate pairing [11]. For most cryptographic applications, the functionality of both are the same. However as can be seen in [21] the computation of the Tate pairing is more efficient than Weil pairing. The Tate pairing first introduced by Frey and Ruck [20].

Tate Pairing Definition: The Tate pairing $T(P, Q)$ where $P, Q \in E[n]$ is a bilinear map from:

$$E[n] \times E[n] \to G_2$$

There is an efficient algorithm for calculating the Tate pairing called Miller algorithm [11]. For understanding the Miller algorithm, it is necessary to introduce some preliminaries.

Suppose we have a supersingular elliptic curve E on a prime field F_p, for example $y^2 = x^3 + 1$ over F_p with $P = 2 \bmod 3$. The following definitions are needed for understanding the Miller algorithm:

Preliminary: The number of points on a supersingular elliptic curve is $P + 1$ over F_p. Suppose $p \in F_p$ with order n divides $p + 1$. Let Q be another point with same order over F_{p^2}. These two points P, Q form a group which we call $E[n]$.

Divisor: A divisor is the summation of some points on curve $E(F_{p^2})$. The formal definition of a divisor is:

$$A = \sum_P a_p(P) \textbf{ where } a_p \in Z \textbf{ ; } P \in E\left(F_{p^2}\right) \textbf{ ; } \sum_P a_p = 0$$

Function: A function f on the curve $E(F_{p^2})$ is a rational function such that:

$$f(x, y) \in F_{p^2}(x, y).$$

Function f for any point P on the elliptic curve E is defined as $f(P) = f(x, y)$. If a function f receives a divisor $A = \sum_{P} a_p$ as an input, the output of the calculation on this divisor is equal to:

$$f(A) = \prod_{P} f(P)^{a_P} \text{ where } \sum_{P} a_P = 0$$

3.1.3.2 Miller Algorithm

After introducing some basic preliminaries, this section discusses about calculation of the Tate pairing with the Miller algorithm. The full explanation of this section can be seen in [11].

Suppose there are two points $P, Q \in E[n]$, for computing $T(P, Q) \in F_{P^2}$. First choose a random value $R_1 \in E[n]$ then let divisor be $A_Q = (Q + R_2) - (R_2)$. The Tate pairing of two points P, Q is:

$$T(P, Q) = F_P(A_Q)^{|F_{p^2}|/2}$$

Lemma 1: For calculation of the Tate pairing, the Miller algorithm provides an efficient recursive algorithm. This algorithm first shows how to compute $f_{b+c}(A_Q)$ from $f_b(A_Q)$, $f_c(A_Q)$, bP , cP and $(b + c)P$.

Let $g_1(x, y) = a_1x + b_1y + c_1$ be the line passing through the points bP,cP and $g_2(x, y) = x + c_2$ be the vertical line passing through the point $(b + c)P$. The following two equations are their divisors:

$$(g_1) = (bp) + (cP) + (-(b+c)P) - 3(O)$$

$$(g_2) = ((b+c)P) + (-(b+c)P) - 2(O)$$

The following algorithm calculates $f_{b+c}(A_Q)$ [11]:

$$f_{b+c}(A_Q) = f_b(A_Q) \cdot f_c(A_Q) \cdot \frac{g_1(A_Q)}{g_1(A_Q)}$$

If $n = b_m b_{m-1} \cdots b_1 b_0$ is the binary representation of n.

Initialization: Let $W = O$, $L = f_0(A_Q) = 1$ and $k = 0$ and

D be the represented algorithm in Lemma 1.

Repetition: For $i = m, m-1, \cdots, 1, 0$ do

1. If $b_i = 1$ then do: Set $L = D(L, f_1(A_Q), W, P, W + P)$, set $W = W + P$ and $k = k + 1$
2. if $i > 0$ set $L = D(L, L, W, W, 2W)$, set $L = 2L$ and $k = 2k$

Result: At the end of the algorithm $k = n$ thus $L = f_n(A_Q)$

3.2 Cryptographic Tools

This section introduces the necessary digital signature methods to preserve the identity of both patients and physicians. First, we introduce the basic concept of the digital signature, then introduce some special digital signatures types required in our implementation.

3.2.1 Digital Signature

A digital signature is a cryptographic tools that shows the authenticity of a digital message. The receiver of a signed message can be sure that this message comes from an authenticated source without any manipulation in middle of the transition. When a certified digital signature is created, it is impossible for sender to deny having signed the message (non-repudiation). Figure 3.1 shows a digital signature schema.

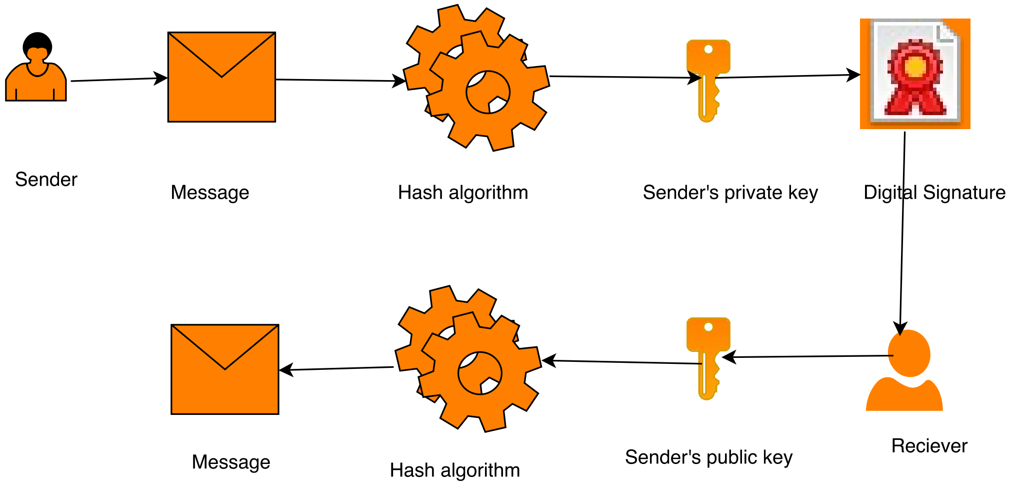


Figure 3.1: Digital signature schema.

As can be seen in Figure 3.1, first the sender uses a hash algorithm to convert a raw message to a binary string, then uses her private key to encrypt the message. At this stage the signed message will be sent to the receiver. At the other side, the receiver uses the public key of the sender to decrypt the signed message. In the final stage, the receiver uses the same hash function algorithm used by the sender to convert back the binary string to the original message.

3.2.2 Proxy Signature

Proxy signature introduced by Mambo et al. in 1999 for first time [30]. In this type of signature, the original signer allows a proxy signer to sign on behalf of her. There are different types of digital signatures introduced like the one in [27]. Zhang et al. in [42] proposed a new proxy signature based on the Boneh–Lynn–Shacham (BLS) short signature from the Weil pairing [12]. This signature has three main steps as follows:

- **Key Generation:** Secret key s is a random number selected from Z_q^*.

  Public key is a tuple (G_1,G_2,q,P,P_{pub},H_2) which $P_{pub} = sP$ and G_1,G_2 are two finite fields.

- **Signing:** For signing a message $M \in \{0,1\}^*$, first message M should be converted to an integer using the hash function H in $P_M = H(M) \in G_1$. The resulting signature is:

  $$S_M = sP_M$$

- **Verification:** For verification, the verifier should check whether the following equation holds.

  $$e(S_M, P) = e(H(M), P_{pub})$$

As mentioned before, the Zhang et al. proxy signature is based on BLS. In this proxy signature, there are two participants called original signer with public key PK_o and a secret key s_o and proxy signer. The proxy signer has PK_p and s_p as public and secret keys respectively.

The original signer and proxy signer should do the following steps in order to produce a valid proxy signature:

1. The original signer produces a warrant $w \in Z_q^*$. This warrant is used for delegation signing ability to the other parties.

2. The Original signer then calculates $So_w = s_o H_2(w)$ and sends w and So_w to the proxy signer.

3. The proxy signer verifies the received input from the original signature by seeing whether the equation $e(So_w, P) = e(H(w), PK_o)$ is held or not. Then calculates $S_w = So_w + s_p H(W)$, where H is a hash function.

4. After that, the proxy signer calculates proxy key S_w, and uses Hess's ID-based signature scheme [22] for producing proxy signature (c_p, U_p) as follows:

   - $r_p = e(P, P)^{k_p}, k_p \in Z_q^*$
   - $c_p = H_1(M \| r_p)$
   - $U_p = c_p S_w + k_p P$

   The following tuple is the signature of the proxy signer:

   $$< M, c_p, U_p, w >$$

5. For verification, the verifier checks that weather the following equation is held or not.

$$c_p = H_1(M\|e(U_p, P)e(H_2(w), PK_o + Pk_p)^{-c_p})$$

3.2.3 Group Signature

Group signature is a special type of digital signatures which allows a member of a group to sign anonymously within a group. This signature can be revoked by the group manager who is responsible for creating this group. In this section, we use the short group signature proposed by Boneh et al. [10]. The process of generating a group signature for physicians is as follows:

- **Keygen(n):** This algorithm takes the number of group members as input n. It chooses an element $h \overset{R}{\leftarrow} G_1 \backslash \{1_{G_1}\}$, $\xi_1, \xi_2 \overset{R}{\leftarrow} Z_P^*$, and a set of two random generators $u, v \in G_1$ such that $u^{\xi_1} = v^{\xi_2} = h$. This algorithm selects $\gamma \overset{R}{\leftarrow} Z_p^*$ and sets $w = g_2^{\gamma}$. Now, the calculation of a private key pair (A_i, x_i) for each user is to randomly derive $x_i \overset{R}{\leftarrow} Z_p^*$ and $A_i \leftarrow g_1^{\frac{1}{(\gamma + x_i)}}$. The group's public key is a tuple $gpk = (g_1, g_2, h, u, v, w)$. Private key of the group manager is $gmsk = (\xi_1, \xi_2)$.

- **Sign($gpk, gsk[i], M$):** The sign algorithm takes public key gpk, a user private key gsk[i], and a message M and outputs a signature σ on M as follow:

$$\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$$

The signer selects exponents $\alpha, \beta \overset{R}{\leftarrow} Z_p^*$. The other parameters of signature are as follows:

$$c \leftarrow H(M, T_1, T_2, T_3, R_1, R_2, R_3, R_4, R_5)$$

the signer chooses $r_\alpha, r_\beta, r_{\delta_1}, r_{\delta_2}$ randomly from Z_p^* and calculates following parameters:

$$R_1 \leftarrow u^{r_\alpha},\ R_2 \leftarrow v_\beta^r,\ R_3 \leftarrow e(T_3, g_2)^{r_x} \cdot e(h, w)^{-r_\alpha - r_\beta} \cdot e(h, w)^{-r_{\delta_1} - r_{\delta_1}},$$

$$R_4 \leftarrow T_1^{r_x} \cdot u^{-r_{\delta_1}}, R_5 \leftarrow T_2^{r_x}.u^{-r_{\delta_2}}$$

$$T_1 \leftarrow u^\alpha, T_2 \leftarrow v^\beta, T_3 \leftarrow Ah^{\alpha+\beta}$$

$$s_\alpha = r_\alpha + c\alpha\ , s_\beta = r_\beta + c\beta,\ s_x = r_x + cx,\ s_{\delta_1} = r_{\delta_1} + c\delta_1$$

- **Verify**(gpk, M, σ): In the verification phase, the verifier recomputes $\tilde{R}_1, \tilde{R}_2, \tilde{R}_3, \tilde{R}_3, \tilde{R}_5$ and checks weather the following equation is held or not.

$$c = H(M, T_1, T_2, T_3, T_4, \tilde{R}_1, \tilde{R}_2, \tilde{R}_3, \tilde{R}_3, \tilde{R}_5)$$

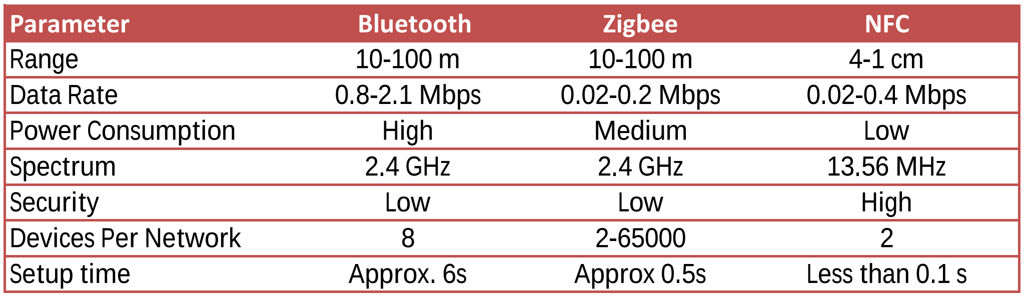
- **Open**($gpk, gmsk, M,\ \sigma$): The Group manager uses the following steps to trace the physician's signature. It takes $gpk = (g_1, g_2, h, u, v, w)$ as the group public key, $gmsk = (\xi_1, \xi_2)$ as the group manager private key together with the message M and the signature $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$. The group manager first verifies that σ is a valid signature on M. Then from the first three parameters (T_1, T_2, T_3), calculates private key of signer $A \leftarrow T_3/(T_1^{\xi_1}, T_2^{\xi_2})$ and then matches it up with private keys $\{A_i\}$ of the group members.

The previous sections explained the short group signature by Boneh and proxy signature by Zhang because in the proposed model in the next chapter they will be used for signing prescriptions by physicians and patients receptively.

3.3 NFC Technology

Near Field Communication (NFC) technology was introduced by Philips and Sony for contactless communication in 2002 [17]. Applications that use NFC range from payment and loyalty applications to access keys for offices and houses . NFC uses a short range wireless communication in order to transmit data between devices [17]. These devices are of three main types: NFC mobile, NFC tag, NFC reader. In order to use services provided by NFC technology, it is necessary to bring two devices near each other. The user first should interact with a smart object. This smart object can be either an NFC tag, NFC reader or another NFC mobile. After interacting with one of these objects with a mobile phone, the received data can be used to initiate another activity in the NFC mobile device. A big advantage of using NFC is storing personal and sensitive data on a secure device that was initially stored on items such as smart card, credit card. Integration of NFC with high power processing mobile phones enables mobile app developers to create innovative apps with benefits like less cash handling, faster transactions, and easier content sharing. Entities involved in NFC include an initiator and a target device. The initiator is responsible for starting and guiding the data exchange process. On the other side, the target device responds to the initiator's requests. There are two modes in NFC protocol: passive mode only active mode. In the active mode, both initiator and target device generate their own Radio Frequency (RF) field to transmit data, but in the passive mode just the initiator creates a field while the target device uses this field to send the data to the initiator.

According to [14], Table 3.1 shows the comparison between different types of wireless communication technologies.



Parameter	Bluetooth	Zigbee	NFC
Range	10-100 m	10-100 m	4-1 cm
Data Rate	0.8-2.1 Mbps	0.02-0.2 Mbps	0.02-0.4 Mbps
Power Consumption	High	Medium	Low
Spectrum	2.4 GHz	2.4 GHz	13.56 MHz
Security	Low	Low	High
Devices Per Network	8	2-65000	2
Setup time	Approx. 6s	Approx 0.5s	Less than 0.1 s

Table 3.1: Comparison of WPAN technologies

In this thesis, NFC will be used at pharmacies by the patient in order to transmit signature of the patient and the physicians to the pharmacist. The pharmacist verifies the patient and physician's signature and if it accomplished successfully, the medication will be delivered to her.

The definitions explained in this chapter will be used for the implementing part in next Chapter. Proxy signature will be used by the patient and her optional proxy signer. The short group signature will be used by the physicians for signing the prescriptions.

Chapter 4

Proposed System

4.1 Model and parties involved

This section demonstrates different entities' roles and their responsibilities in the proposed e-prescribing system. For illustrating these entities visually and their responsibilities, a use case diagram of whole system will be provided in figure 4.1. The following paragraphs introduces involved parties in the system:

- **Group Manager** GM: This entity is responsible for creating a group of physicians and generate public and private keys for them. She can trace the signature of the group members in case of any future dispute.

- **Patient** P: The patient P is the main entity for whom the prescription is issued. The patient needs to sign the prescription in order to show her consent to the process. The patient has the option to collect their drugs by another trusted party such as agents and family members in case of sickness. This person is called proxy signer, which has been explained in chapter 3. The patient uses

NFC technology on her smartphone to prove the identity instead of showing an ID card.

- **Physician** DR: This entity is created by the group manager and has its own public and private keys. The physician is able to sign the prescriptions with her private key. The physicians reviews the drug history of the patient checks reactions among drugs for preventing any conflicts between them. The proposed system notifies the physician for these type of conflicts before prescribing.

- **Insurer** I: Insurer entity is responsible for paying for medications fees. This entity receives the prescription and the physicians's signature from the physician along with the patient's information. It verifies the signature and calculates the bill for providing it to the pharmacy.

- **Proxy Signer** Pr: In the case that the patient P is not able to collect her medicine, this identity accepts the proxy signature and signs on behalf of the main patient to be able to collect the medications.

- **Pharmacy** PH: This entity receives the signatures of both patient and physician and verify the validity of them, then delivers the medicine to verified patient.

4.2 Model

Figure 4.1 shows the proposed e-prescribing use case diagram and parities involved in this system. The different stages of the implementation will be showed in more details in Architecture section 4.3.

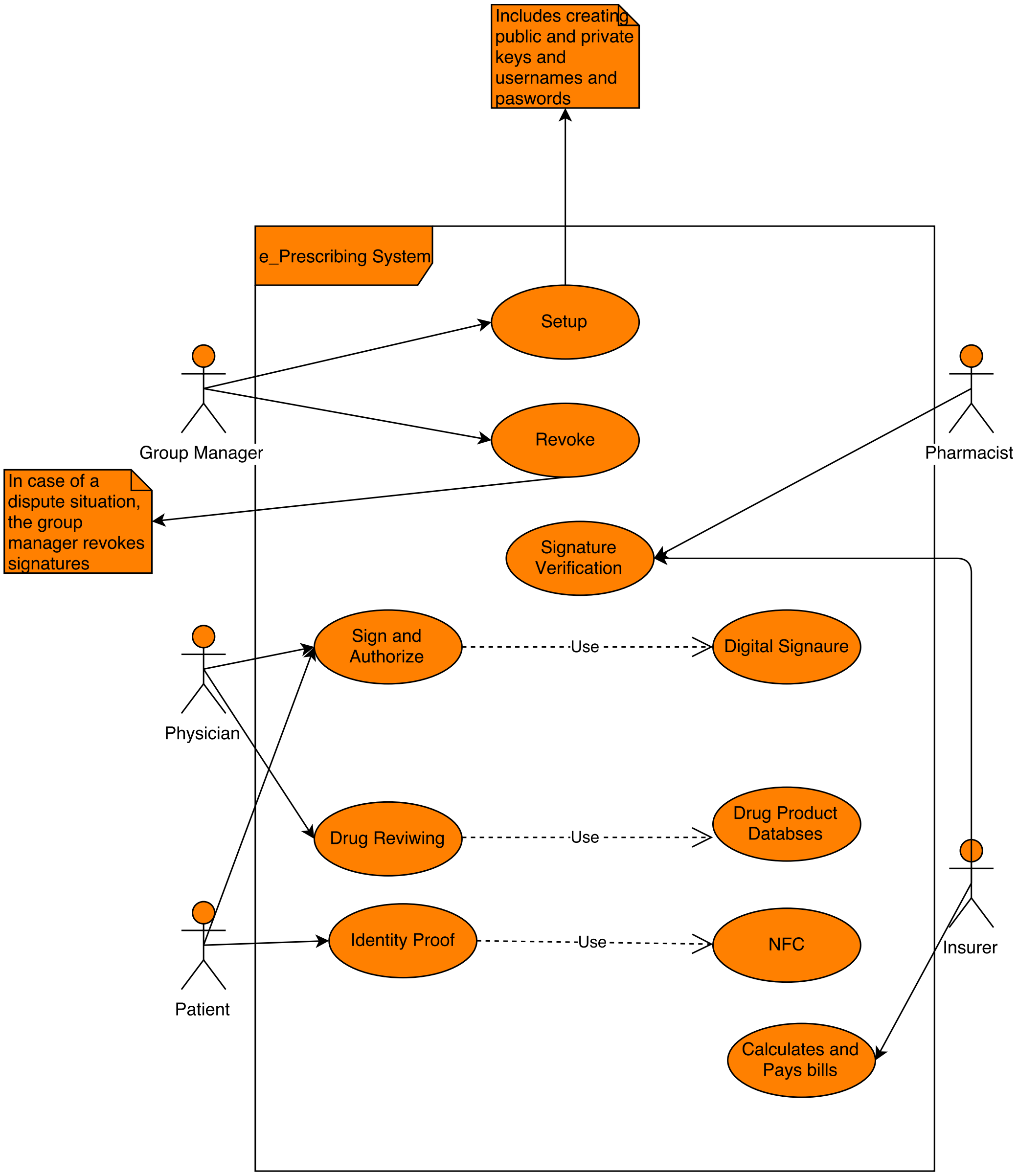


Figure 4.1: e_Prescription use case diagram.

4.2.1 UML Diagram of Involved Parties

UML is a modelling language to describe the architecture of a system. This chapter discusses the architecture of the proposed e-prescription system and the relation between all involved parties using various UML diagrams. Figure 4.2 shows the object diagram of all parties in the system.

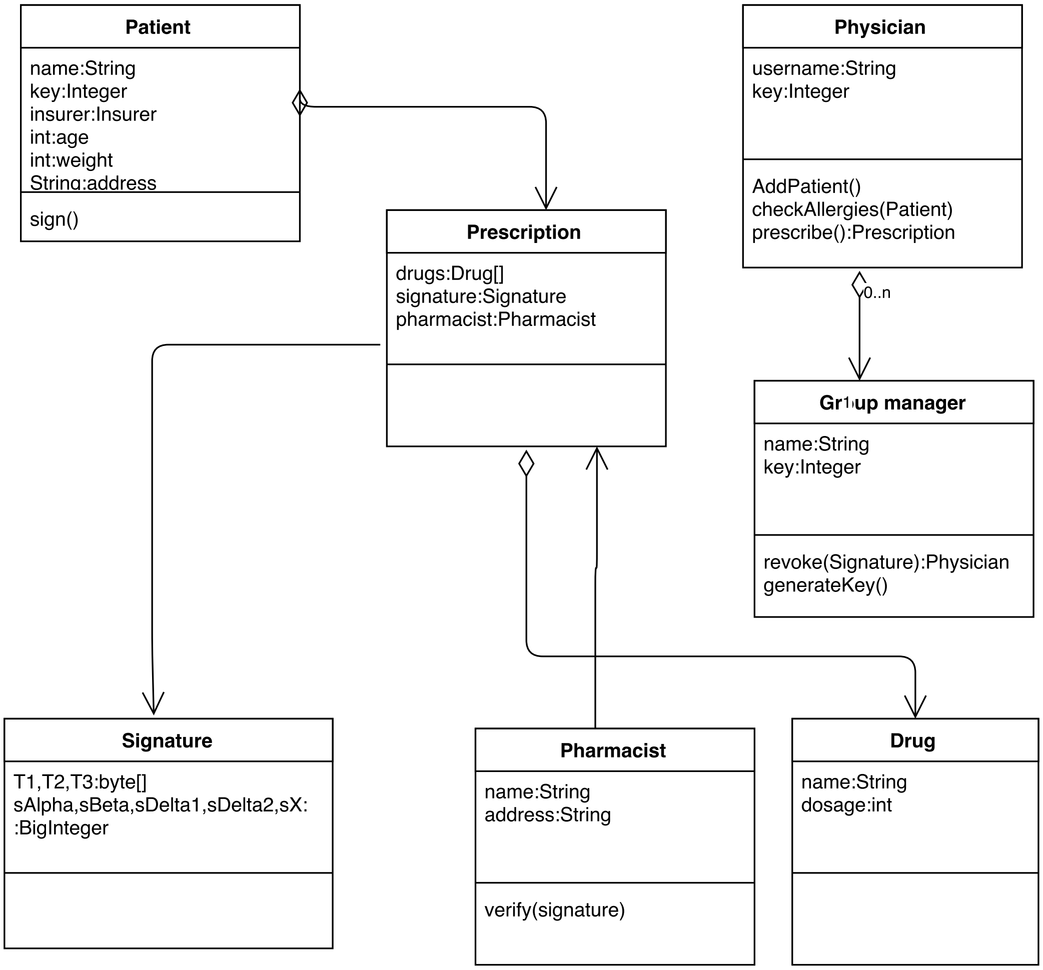


Figure 4.2: Class Diagram of the e-prescription System

As can be seen in Figure 4.2, the proposed system has 8 main entities. These entities have some methods and properties in their definitions. The following paragraphs will briefly explain these entities and their functions:

- Patient

  - Properties: First name, last name, user name, key, age, sex, weight, address, and phone.
  - sign(Prescription): In this methods, the patient receives a prescription object as the input and signs it with her private key producing a digital signature.

- Physician
  - Properties: First name, last name, user name, private key
  - Prescribe(Prescription): In this method, the physician receives a prescription as an input object, then signs it with her private key.
  - AddPharmacy(): This method adds a new pharmacy to the list of eligible pharmacies.
  - AddPatient(): In this method, the physician creates a new patient entity based on provided information from the patient.

- Drug: This entity represents the drug as our basic object which will be used in the prescription object. The properties of the drug object are:
  - Name, Dosage

- Prescription: This object represents the prescription that needs to be signed by both the physician and the patient.
  - Properties: List of Drugs, Date, Signature.

- Signature: This entity is our signature object which had its properties introduced in chapter 3.

- Pharmacist
  - Properties: user name, address, phone.
  - Methods: Verify(): this method will be used for checking the validity of the prescription's signature.
- Group manager: This entity is responsible for creating the physician's group and revoking the signature in a dispute situation.
  - Properties: User name, Private key
  - Methods: addPhysician(), revokePrescription()

4.3 Process Flow

In this section, the architecture and each step in the proposed e-prescription system will be discussed. The following steps are required during the whole process of the e-prescription system.

4.3.1 Setup

To setup the system for first time, it is necessary to do the following steps.

$S_0 : GM \longrightarrow Dr(Private, public\,Keys, ID)$

$S_1 : P \longrightarrow I(Bank\,Information, Health\,Plan)$

$S_2 : P \longrightarrow Pr(\omega)$

In S_0, the group manager generates public and private keys for the physician which is a certified member of the physician's group. With this private key, the physician will be able to sign the prescription. Also a ID number will be created for each physician to be connected with their public keys. In case of a dispute situation, The group manager revokes physician's signature with her master key. In S_1, the patient sends the health plan which is interested to be signed up for it and also her banking information to set up a new account. In S_2, the patient sends a warrant ω to proxy signer for establishing a proxy signature. The process of creating a proxy signature has explained in chapter 3.

4.3.2 Execution

After the setup phase in the beginning, the next phase which is the execution phase is as follows.

$S_3 : P \longrightarrow DR(Insurance\,ID, Pharmacy\,Name)$

$S_4 : DR \longrightarrow P(S_1(m), m,\ session\,key\,K)$

$S_5 : DR \longrightarrow PH(S_1(m),\ m,\ session\,key\,K, ID\,number)$

$S_6 : DR \longrightarrow I(m,\ S_1(m),\ Patient\,ID,\ K)$

$S_7 : I \longrightarrow PH(session\,key\,K, Bill)$

$S_8 : P \longrightarrow PH(S_2(m),\ session\,key\,K)$

$S_9 : PH \longrightarrow P(Drugs)$

In S_3, the patient gives her identification information like insurance ID number to the Doctor. The patient also clarify her pharmacy of choice for the physician. In S_4, after examination of the patient by the physician, the physician checks allergies of the patient in her record. The physician then prepares the prescription and signs it. The signature will be prepared in $S_1(m)$ and sent it to the patient smartphone application. This signature is a type of group signature introduced in [10].The physician also creates a session key k for each visitation by the patient. The session key will be used at pharmacy to identify the patient. The plain message is in m and will be delivered to the patient in S_4.

In S_5, the physician,after accomplishing diagnosis, sends her signature, prescription, session key and her identification number. Since the public keys,needed for verifying the signature, are published under these identification numbers, the pharmacist uses this number to verify the signature without revealing the real identity of the physician. In S_6, the physician sends the prescription, her signature and the real identity of patient to the insurance company to prepare the bill. In the next step S_7, the insurance company verifies the physician's signature for making sure about the verifiability then prepare the bill regarding to the identity of the patient and the prescription received from the physician. In S_8, upon visiting the pharmacy, the patient uses her smartphone's NFC to transfer the session key and her signature to the pharmacist. The pharmacist verifies the signature of the patient. If succeed the bill corresponded with the session key has to be paid by the patient before dispensing the medications . After paying the bill, the pharmacy dispenses medication in S_9.

4.3.3 Revocation

After step 4, the signature $S_1(m)$, and message m will be inserted into the database as a prescription. In a dispute situation when the group manager receives a prescription containing $(S_1(m), S_2(m),\ m,\ K)$, compares both physician's signatures to look it up in the database then uses the revocation method introduced in chapter 3 to figure out the signer of the disputed prescription.

4.4 Sequence Diagram

Figures 4.3 and 4.4 show tasks of the physician and the patient respectively. These figures are adopted the sequence diagram to explain the architecture's steps, introduced above, in a numerical order.

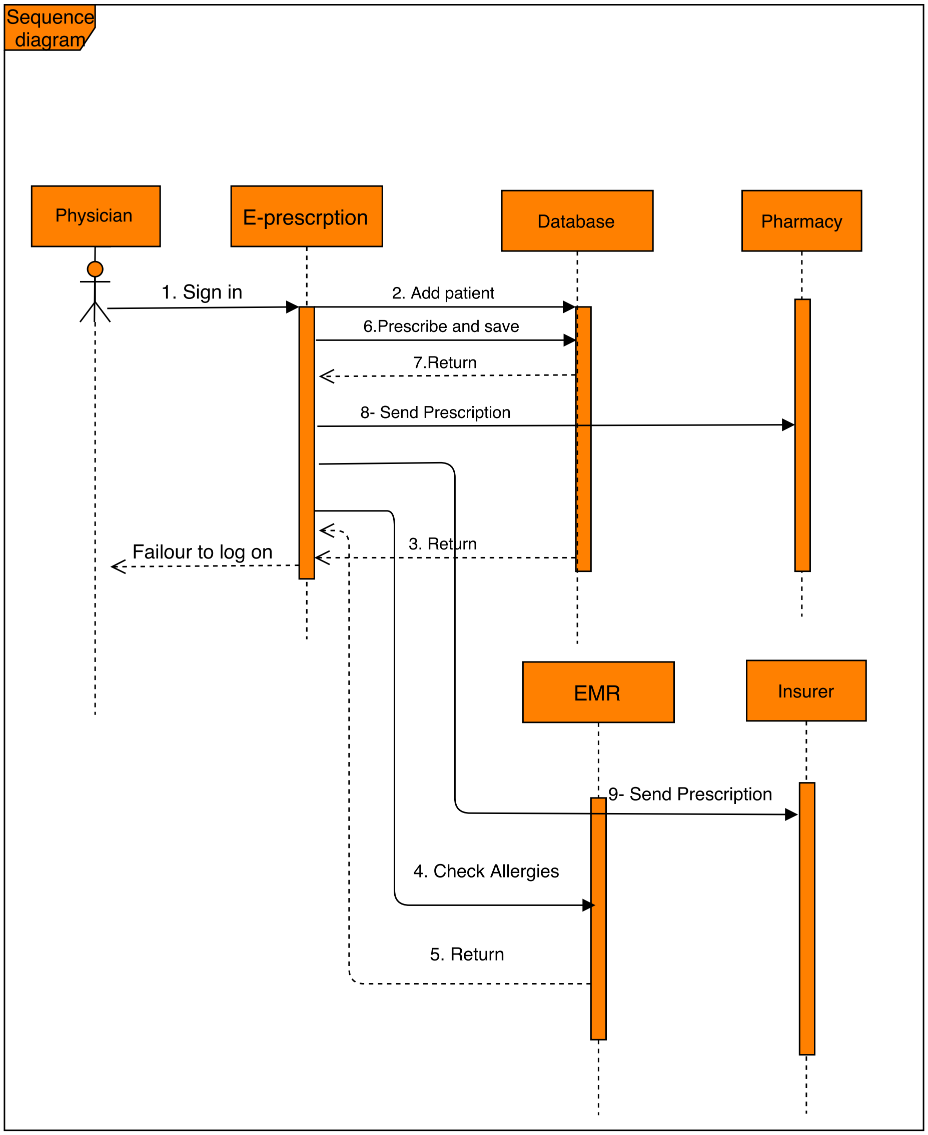


Figure 4.3: Physician sequence diagram.

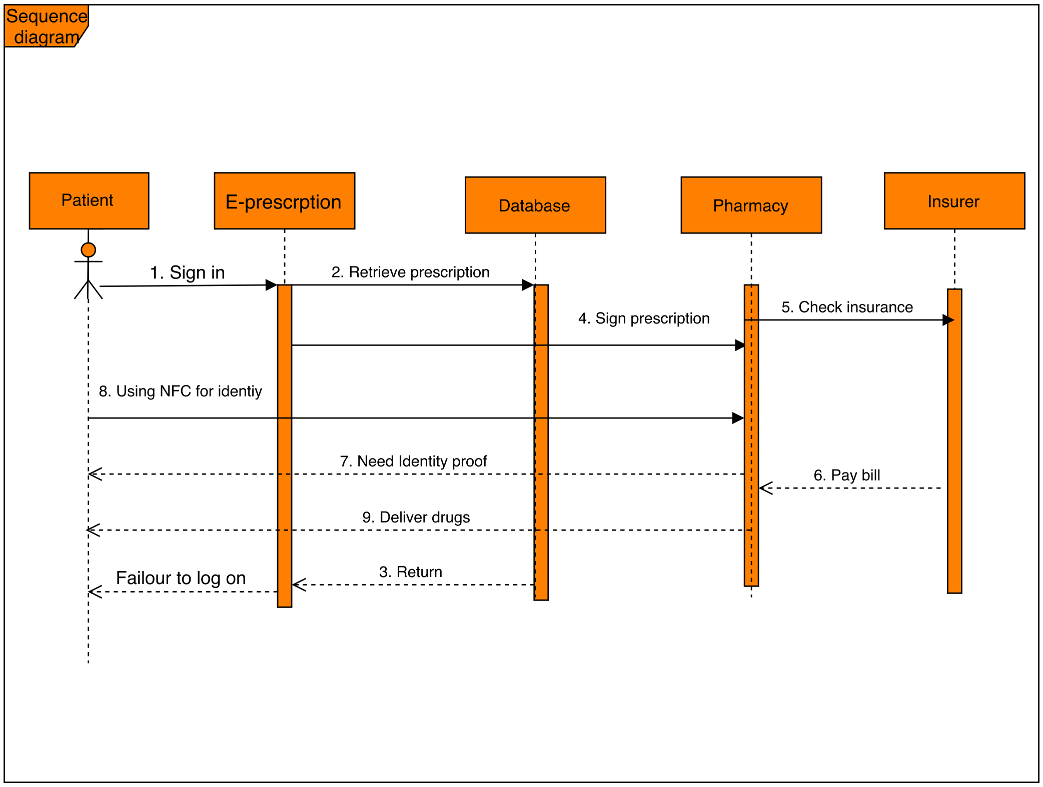


Figure 4.4: Patient sequence diagram.

4.5 Information Security in E-prescription System

In order to make the proposed e-prescription system secure, like any other digital system, it is necessary to have the following requirements:

- **Secrecy:** Secrecy means that not revealing the sensitive information to a non-authorized party. This requirement will be met in the proposed system because all parties in the proposed system have a level of access to the sensitive information. Every party has their username and password in order to login to the system. For instance the pharmacy entity is just allowed to see the prescription content while the physician is allowed to see the patient's information and has access to add or edit patients' information. The group manager has higher level of access so that she is able to revoke the signatures. Therefore, all of the users have access to their own role-related information ensuring non-authorized recipients can not access to sensitive information.

- **Integrity:** Integrity of data must be ensured all the times, therefore users will be ensured that their data can not be interfered or changed during the process of e-prescribing. For instance, while transmuting data to the pharmacy, the user must be ensured nobody can change the dosage of their drugs. This requirements is satisfied because of using digital signatures in the system. If any contents of signatures and/or messages are changed in transition the signature will not be verified properly, on the other end.

4.6 Complexity Assumption

The complexity of signatures used in the proposed system is based on the Strong Diffie-Hellman problem. According to [16], if g, g^{α} and g^{α^d} are given for a positive divisor h of order $p-1$, the secret $\alpha \in Z_p$ can be calculated in $O(\log p.(\sqrt{\frac{p}{h}} + \sqrt{h}))$ group operations and using $O(max \left\{ \sqrt{\frac{p}{h}}, \sqrt{h} \right\})$. In the proposed system because p is equal to 256 bits and h is 15 bits, finding the private key takes $O(2^{126})$ group operations and $O(2^{120})$ memory operations.

4.7 Conclusion

In this chapter, the model and process flow has been introduced. These are important to have an idea and perspective of the system for implementing it on real platforms. The next chapter will use these models and process flow to implement the system on both web and mobile platforms.

Chapter 5

Implementation

This section discusses tools which have been used in implementation part. These tools consist of programming languages like Java, and programming frameworks such as Spring [6], Spring Security, Bootstrap, and Hibernate. Using of these programming frameworks helps us to have a more reliable and testable software implementation. In programming frameworks, developers are more focused on the unique requirements of their application and less concerned about syntax and programming details. The following paragraphs explain these tools in more detail.

5.1 Java

Java is the most famous existing object-oriented programming language. For the most parts of the implementation, Java has been used. Because Java is an object-oriented programming language, it is easy to define every involved party as an object and it will make it easy to add new parties in the future too. Java also is a platform-independent language which makes it runnable on every platform.

5.2 Java Pairing Cryptography Library

JPBC is an open source library developed by Ben Lynn [13] to execute mathematical operations in pairing based cryptography. JPBC works with both bilinear mapping and multilinear mapping. In this project, our focus is on bilinear pairing which has two main properties:

1. **Bilinearity:** For all $a, b \in Z_r$ the equation $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$ holds

2. **Non-degeneracy:** $e(g_1, g_2) \neq 1$

For using this library in order to deriving pairing between two elements g_1 and g_2, it is necessary to introduce some steps to initiate preliminaries as follows:

1. **Bilinear Pairing Parameters Generators**

   The proposed e-prescription system uses type A pairing. This type of pairing is based on the curve $y^2 = x^3 + x$ over a field F_q for prime $q = 3 \bmod 4$. This pairing is symmetric because both elements that constructs the pairing will be derived from G_1 and G_2, each of which is a group of points over F_q. The following code shows the parameters needed for generating pairing for the type A.

```
int rBits = 160;
int qBits = 512;
// JPBC Type A pairing generator...
ParametersGenerator pg = new TypeACurveGenerator(rBits,
   qBits);
// PBC Type A pairing generator...
ParametersGenerator pbcPg = new PBCTypeACurveGenerator(
   rBits, qBits);
```

It shows that order r is an integer with 20 bits length and mod q is 512 bits digit. After running this code and generating related parameters, we have parameters in a separate file named param.properties with the following content:

```
type a
q=8780710799663312522437781984754049815806883199414208
211028653399266475630880222957078625179422662221423155
85876958231745927771336731748132492512999822479 1
h=12016012264891146079388821366740534204802954251
311822919615131047207289359704531102844802183906537786
776
r=730750818665451621361119245571504901405976559617
exp2 159
exp1 107
sign1 1
sign0 1
```

r is a Solinas prime number with format $2^a + -2^b + -1$ for some integers $0 < b < a$. h is also another integer satisfying $q + 1 = r * h$.

2. **Initializing Pairing:** In JPBC there is an interface called Pairing providing

some helpful methods and tools. For instantiating an object from this interface, JPBC has a Pairing factory function. This pairing factory receives parameters q, r, and h, which are generated in the previous section, then calculates pairing between two elements. Suppose that paring parameters are available in a file named param.properties, the following code creates a Pairing object for calculating pairing between two elements on the base curve:

```
Pairing pairing = PairingFactory.getPairing("params.
    properties");
```

3. **Algebraic Structure:** This section explains how to calculate the pairing between two elements from the pairing. For this purpose, first we should obtain a field which contains the elements. The following code shows how to derive essential fields to produce the necessary elements.

```
Field Zr = pairing.getZr();
Field G1 = pairing.getG1();
Field G2 = pairing.getG2();
Field GT = pairing.getGT();
```

The number of available fields depends on the degree of the pairing class. In particular, number of these structures equals to $degree + 1$. The degree can be calculated by *getDegree()* method.

```
int degree = pairing.getDegree();
```

4. **Apply Pairing:** After calculating fields from the pairing, it is easy to derive the elements laying on the curve.

```
/* Create a new uninitialized element. */
Element e1 = G1.newElement();
/* Create a new uninitialized element. */
Element e2 = G2.newElement();
//Applying pairing between two elements e1,e2
Element out = pairing.pairing(e1,e2);
```

5.3 Spring MVC

Spring MVC framework handles the Model-View-Controller architecture of a Java web-based application. This framework helps an application have loosely coupled classes. In this type of application, components do not depend on each other heavily, making it easier to test each component independently. Following are three main independent components of a loosely coupled application.

- **Model**

  The Model represents the part of an application which is dealing with data like those classes that handle retrieving or storing requests from a database.

- **View**

  This part handles the display of an application. It renders the model objects and shows them in HTML output which browsers can translate them to visual objects.

- **Controller**

  Controllers are the main part of the application which handles all of the user requests. Depending on the received requests, controllers create an appropriate model and redirect these requests to desired views.

5.3.1 Spring MVC Dispatcher

In the core of the dispatcher servlet, there is a class called Dispatcher Servlet responsible for all incoming HTTP requests from users. The workflow of the Dispatcher Servlet looks like the following diagram:

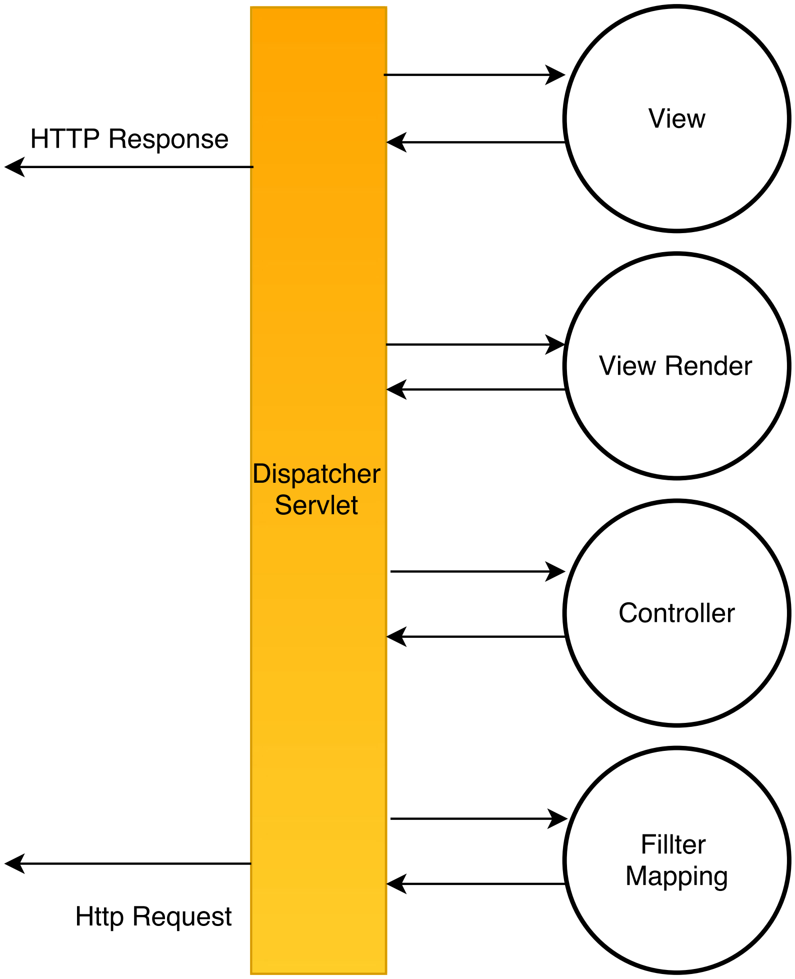


Figure 5.1: Dispatcher Servlet

As can be seen in the Figure 5.1, all HTTP requests coming from users should pass from the dispatcher servlet. The lifecycle of a single HTTP request in a Spring MVC- based system is as follows:

1. When a HTTP request is received at the Dispatcher servlet's input gate, it asks from filter mapping to redirect to the appropriate controller.
2. After the dispatcher sends a request to the correct controller, the intended services will be called. This service could be a pushing to the database or a mathematical operation. Finally, the dispatcher servlet renders a view reflecting what is being done in the backend process.
3. Once the dispatcher receives a view from controller, it will be redirected to an appropriate view.

5.3.2 Spring MVC Configuration

In order to configure what type of requests should be served by the DispatcherServlet, it is necessary to declare a mapping in web.xml. This file is responsible for defining the structure of a J2EE web application. In the following example, all requests that end with *.html* will be served by the Dispathcer Servlet.

```
<servlet>
<servlet-name>MVCDispatch</servlet-name>
<servlet-class>
org.springframework.web.servlet.DispatcherServlet
</servlet-class>

</servlet>
<servlet-mapping>
<servlet-name>MVCDispatch</servlet-name>
<url-pattern>*.htm</url-pattern>
</servlet-mapping>
</web-app>
```

5.3.3 Spring MVC Controller

As mentioned in the previous section, controllers are used for serving http requests incoming from users. In the spring MVC frameworks, these controllers will be defined by *@Controller* annotation at the beginning of every controller class. Another important annotation is *@RequestMapping(URL)*, which can be used for entire class or a specific method. This annotation shows that the specific method serves address with the URL format. The example below shows that all URLs which contain *admin* word will be handled by adminPage() method.

```
@RequestMapping(value = "/admin**", method = RequestMethod.GET)
public ModelAndView adminPage() {
ModelAndView model = new ModelAndView();
model.addObject("title", "Welcome to e-prescription system");
model.setViewName("admin");
return model;
}
```

5.4 Spring Security

Spring security is an authentication and access control framework. The purpose of this framework is for securing spring application. This framework allow different roles to authenticate themselves to the system and be able to log in to to the system. Authorization is the process of checking the privilege of users to restrict their access to the content of applications based on their roles. Spring security retrieves users' information like user names and passwords in order to allow users to accomplish their task depending on their roles.

5.4.1 Spring Security Configuration

Spring Security can work with databases such as mySQL to retrieve the user's credential information. As can be seen in this reference [2], the following code sets access Spring Security to the database.

```
<bean id="dataSource"
class="org.springframework.jdbc.datasource.DriverManagerDataSource">
<property name="driverClassName"value="com.mysql.jdbc.Driver" />
<property name="url" value="jdbc:mysql://localhost:3306/test" />
<property name="username" value="root" />
<property name="password" value="password" />
</bean>
```

To force Spring Security to redirect requests with special formats which, to the login page, the configuration will be like below:

```
<intercept -url pattern ="/admin*" access ="hasRole(’ROLE_ADMIN’)"/>
<intercept -url pattern ="/phys*" access ="hasRole(’ROLE_PHYS’)"/>
<intercept -url pattern ="/pharm*" access ="hasRole(’ROLE_PHARM’)"/>
```

5.5 Hibernate

Hibernate is an Object Relational Mapping(ORM) framework developed by Red Hat [7]. ORM is a programming method to transform objects in object-oriented programming languages like Java to a relational database like MySQL. Hibernate facilitates data mapping from Java objects to databases’ tables. Hibernate also provides data queries to store and retrieve data to/from databases. This framework is an open source framework licensed under GNU. Hibernate uses XML files to translate Java objects to tables in a database. Properties in Java objects would be rows in the database. For example, following XML file represents the user object.

```
<hibernate-mapping>
```

```
<class name="User" table="User">
<meta attribute="class-description">
This class contains the employee detail.
</meta>
<id name="id" type="int" column="id">
<generator class="native"/>
</id>
<property name="firstName" column="first_name" type="string"/>
<property name="lastName" column="last_name" type="string"/>
<property name="role" column="role_name" type="string"/>
</class>
</hibernate-mapping>
```

5.6 Bootstrap

Bootstrap is a framework for designing responsive and mobile-friendly views for web-based applications. The core of Bootstrap is based on HTML, CSS, JavaScript. It prepares some templates such as forms, buttons, tables, image carousel, etc. Bootstrap is developed by Mark Otto and Jacob Thornton at Twitter [18].

5.7 Database Design

As can be seen in the Figure 5.2, there are following relationships between objects:

- **Physician ↔ Patient:** There is a many to many relationship between the physician and the patient. Every physician can have one or many patients and vice versa, every patient can have one or more physicians.

- **User ↔ User Role:** The relationship between these two objects is one to many. Every user can have one or more user roles. For instance, the group manager can be a physician as well.

- **Prescription ↔ Drug:** Every prescription can include one or more drugs, and every drug can appear in one or more prescriptions. Therefore, the relationship between them is many to many.

- **Prescription ↔ Signature:** Every prescription, depending on its content, can have a unique signature. So, the relationship between them is one to one.

- **Prescription ↔ Pharmacy:** This relationship, as can be seen in Figure 5.2, is a many to one relationship. Several prescriptions can be sent to one pharmacy.

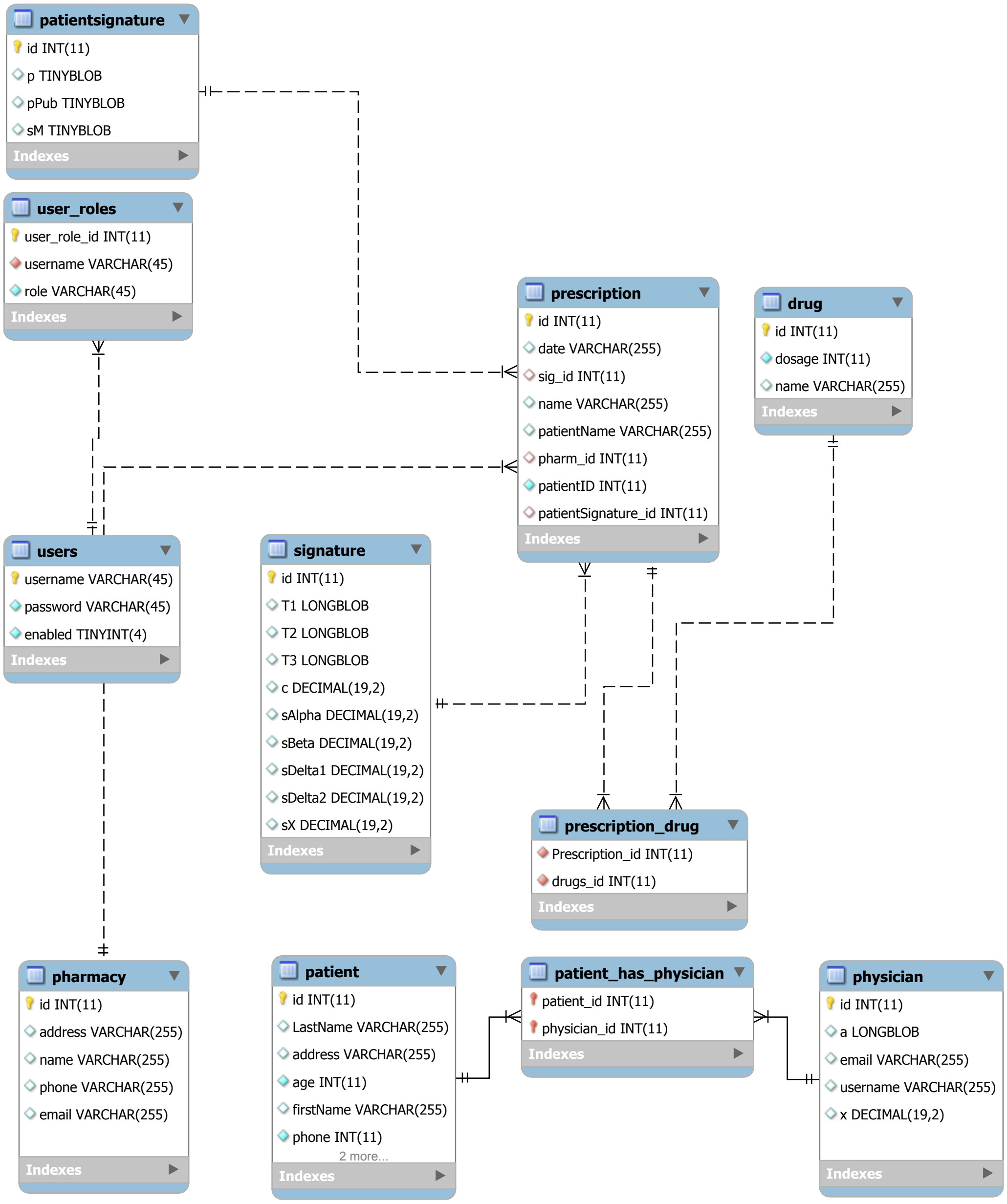


Figure 5.2: Database Schema.

5.8 Chapter Conclusion

The reason for adopting these frameworks for implementation here is because they make the application suitable for the enterprise level in the future. Enterprise applications require implementation tools that be able to handle high number of requests simultaneously. Also, handling authentications are vital for the system as every role has their own responsibilities and restrictions. Therefore, there should be an authentication mechanism in place, here the Spring Security, to handle this part.

Chapter 6

Experimental Results

6.1 Performance Analysis

In this chapter, the performance of three main entities, the group manager, physician, and verifier will be monitored. This performance checking is based on time and number of bits of the elliptic curve used in our implementation. Evaluating time is important because finding subgroups of points in an additive group and subsequently finding the generator on the elliptic curves are very time consuming. As mentioned in Chapter 3 on every elliptic curve over F_p there exist p points. Therefore, the order of points are $O(2^n)$ which n is the number of bits in the mod.

Hardware configurations that these tests has been accomplished on them are as follows:

- MacBook Pro (OS X El Capitan)
- Processor: 2 GHz Intel Core i7

- Memory: 8 GB 1600 MHz DDR3

6.1.1 Group Manager Performance

As can be seen in Figure 6.1, key generation time is rising per number of physicians. The reason that justifies this behaviour is that the server needs to find a private key A such that $A^{x+\gamma} = g_1$ where $x, \gamma \in Z_p$ are random numbers and g_1 is a generator. Calculation of this A for every physician takes some processing time.

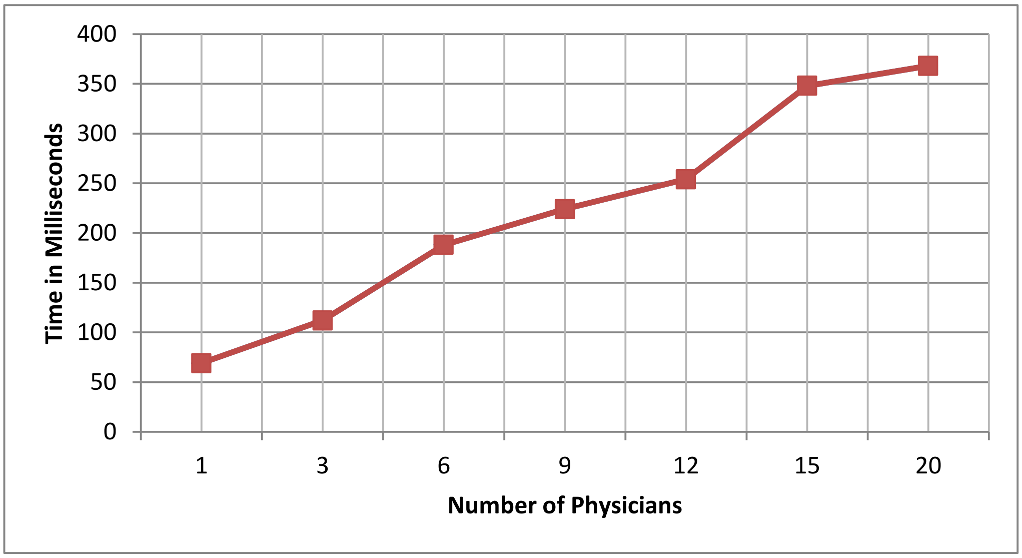


Figure 6.1: Generate Keys for Physicians.

In the next Figure, 6.2, the horizontal data axis shows the bit number of the *mod*. This *mod* has been discussed in Chapter 3. The elliptic curve that has been used in the implementation is a type A elliptic curve with mod q. The mod can be changed to any arbitrary length. In this performance analysis, the range is from 50 bits to 512 bits. As can be seen in Figure 6.2, the revocation time does not change during this test.

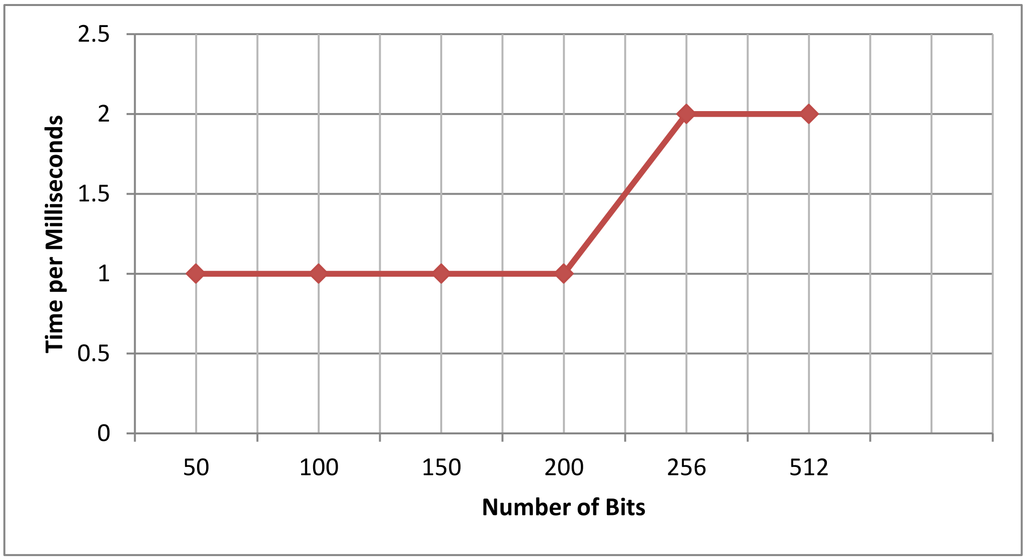


Figure 6.2: Revoking Time per Number of Bits.

6.1.2 Signer Performance

Figure 6.3 shows the relationship between signing time and the mod's length per bit. The signing process consists of ten multiply functions and eleven power functions. By growing the number of bits, the elements on the elliptic curve gets more lengthy and complex. So, this calculations consume more processing time and as can be expected, the time will be growing per number of bits.

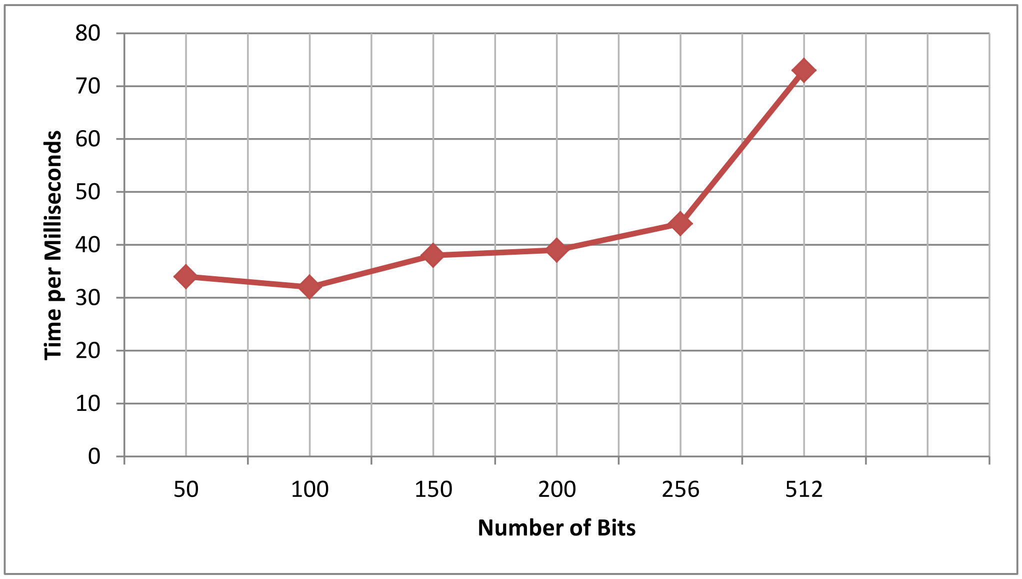


Figure 6.3: Signing Time per Number of Bits.

6.1.3 Verifier Performance

Another performance test, which can be seen in Figure 6.4, is a linear chart between the number of bits and verification time parameters. For verification, the server derives all R_1 to R_5 variables again at the verifier side and calculates the challenge c, as explained in Chapter 3. The server compares this challenge with the existing challenge in the signature of the physician. If they are equal, then verification is done. This process consists of ten multiplying operations, eleven power operations, and two mod inverse calculations. By growing the number of the mod's bits, this calculation consumes more processing time, because elements on the elliptic curve get more complex.

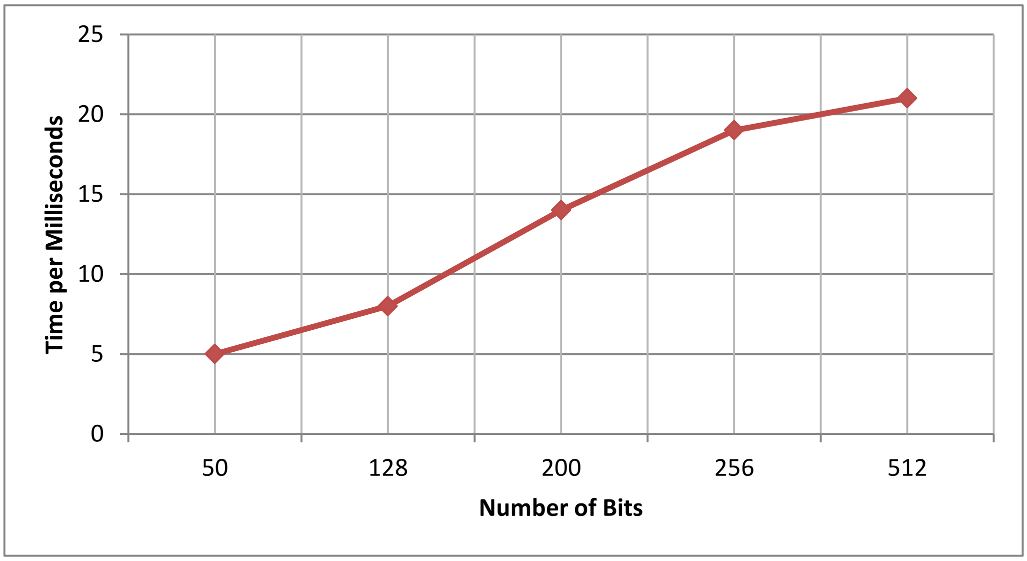


Figure 6.4: Verification Time per Number of Bits.

Chapter 7

Conclusion and Future work

7.1 Conclusion

This study have introduced a secure e-prescribing system based on NFC technology. The main goal of this system is to protect the identity of both patients and physicians. In this system, pharmacies and drug companies are not able to track prescriptions' content and use their data to promote their products or other marketing purposes that has negative affects on physicians' decisions. In [41], an e-prescription system had been proposed based on smart-card along with a group signature scheme. One improvement that this proposed system has over the other systems is using NFC technology and smartphones. NFC Technology is a radio communication in short distance has many applications nowadays, like contactless payment. In this thesis this technology has been adopted for identity proof of the patient. Using NFC is also more secure than using a smart-card. Another advantage of using smartphones instead of smart-card is the patient's ability to track the history of prescriptions at all times. This thesis also used pairing based cryptography which has smaller key size with equivalent security in compare to previous works making it suitable to use with handheld devices with power and processing constraint.

7.2 Future Work

Testing of the proposed system with real data and real parties such as patients, physicians, group manager, insurer companies, and pharmacies will be the future direction of this thesis.

Another case that can be seen as a future work is ability to check patterns of drugs by the group manager in different areas. For doing this, the group manager checks all prescriptions' contents in order to find repetitive patterns in prescriptions. This can help the group manager to find possible outbreaks in specified areas to prevent these outbreaks to other areas as soon as finding these patterns. This task should be done without revealing the identity of physicians. Since the group manager is responsible for doing this task and at the same time she is able to revoke prescriptions in dispute situations, there should be some restrictions to prevent the group manager from revealing the identity of physicians while she checks for outbreaks in specific locations. Testing of this system with real data might need to use some data from other sources like EMR and EHR. Therefore, integration this e-prescribing system with EMR and EHR is a future work.

Because the insurer entity has some special and commercial policies, and it has no specific impact on the main purpose of this thesis, its implementation has been left as future work. Integration and comply with NCPDP standards for communication and transferring prescriptions between the entities with other healthcare providers is possible future work.

Bibliography

[1] Safe e-prescribing: A primer for practices. `http://www.physicianspractice.com//`.

[2] Patrick Kierkegaard. E-prescription across europe. *Health and Technology*, 3(3):205–219, 2013.

[3] John T James. A new, evidence-based estimate of patient harms associated with hospital care. *Journal of patient safety*, 9(3):122–128, 2013.

[4] Martin A Makary and Michael Daniel. Medical error—the third leading cause of death in the us. *BMJ*, 353:i2139, 2016.

[5] Timothy S Lesar, Laurie Briceland, and Daniel S Stein. Factors related to errors in medication prescribing. *Jama*, 277(4):312–317, 1997.

[6] Giampaolo P Velo and Pietro Minuz. Medication errors: prescribing faults and prescription errors. *British journal of clinical pharmacology*, 67(6):624–628, 2009.

[7] David Orentlicher. Prescription data mining and the protection of patients' interests. *The Journal of Law, Medicine & Ethics*, 38(1):74–84, 2010.

[8] Charles A Walton. Portable radio frequency emitting identifier, May 17 1983. US Patent 4,384,288.

[9] Samuel D Warren and Louis D Brandeis. The right to privacy. *Harvard law review*, pages 193–220, 1890.

[10] Safeguarding personal health information. `https://www.ipc.on.ca/`.

[11] Joseph W Thompson, Sathiska D Pinidiya, Kevin W Ryan, Elizabeth D McKinley, Shannon Alston, James E Bost, Jessica Briefer French, and Pippa Simpson. Health plan quality-of-care information is undermined by voluntary reporting. *American journal of preventive medicine*, 24(1):62–70, 2003.

[12] Stanley Trepetin. *Privacy in context: the costs and benefits of a new deidentification method.* PhD thesis, Massachusetts Institute of Technology, 2006.

[13] Khaled El Emam, Fida Kamal Dankar, Romeo Issa, Elizabeth Jonker, Daniel Amyot, Elise Cogo, Jean-Pierre Corriveau, Mark Walker, Sadrul Chowdhury, Regis Vaillancourt, et al. A globally optimal k-anonymity method for the deidentification of health data. *Journal of the American Medical Informatics Association*, 16(5):670–682, 2009.

[14] National council for prescription drug programs. `https://healthit.ahrq.gov/key-topics/ncpdp//`.

[15] Simon Liu, Wei Ma, Robin Moore, Vikraman Ganesan, and Stuart Nelson. Rxnorm: prescription for electronic drug information exchange. *IT professional*, 7(5):17–23, 2005.

[16] Computerized provider order entry. `https://psnet.ahrq.gov/primers/primer/6/computerized-provider-order-entry/`.

[17] David W Bates, David J Cullen, Nan Laird, Laura A Petersen, Stephen D Small, Deborah Servi, Glenn Laffel, Bobbie J Sweitzer, Brian F Shea, Robert Hallisey, et al. Incidence of adverse drug events and potential adverse drug events: implications for prevention. *Jama*, 274(1):29–34, 1995.

[18] David C Radley, Melanie R Wasserman, Lauren EW Olsho, Sarah J Shoemaker, Mark D Spranca, and Bethany Bradshaw. Reduction in medication errors in hospitals due to adoption of computerized provider order entry systems. *Journal of the American Medical Informatics Association*, 20(3):470–476, 2013.

[19] Johanna I Westbrook, Melissa T Baysari, Ling Li, Rosemary Burke, Katrina L Richardson, and Richard O Day. The safety of electronic prescribing: manifestations, mechanisms, and rates of system-related errors associated with two commercial systems in hospitals. *Journal of the American Medical Informatics Association*, 20(6):1159–1167, 2013.

[20] Giuseppe Ateniese and Breno de Medeiros. Anonymous e-prescriptions. In *Proceedings of the 2002 ACM workshop on Privacy in the Electronic Society*, pages 19–31. ACM, 2002.

[21] David Chaum and Eugène Van Heyst. Group signatures. In *Advances in Cryptology—EUROCRYPT'91*, pages 257–265. Springer, 1991.

[22] Yanjiang Yang, Xiaoxi Han, Feng Bao, and Robert H Deng. A smart-card-enabled privacy preserving e-prescription system. *Information Technology in Biomedicine, IEEE Transactions on*, 8(1):47–58, 2004.

[23] Chien-Lung Hsu and Chung-Fu Lu. A security and privacy preserving e-prescription system based on smart cards. *Journal of medical systems*, 36(6):3637–3647, 2012.

[24] Adi Shamir. Identity-based cryptosystems and signature schemes. In *Advances in cryptology*, pages 47–53. Springer, 1985.

[25] Dan Boneh, Xavier Boyen, and Hovav Shacham. Short group signatures. In *Advances in Cryptology–CRYPTO 2004*, pages 41–55. Springer, 2004.

[26] Dan Boneh and Matthew Franklin. Identity-based encryption from the weil pairing. *SIAM Journal on Computing*, 32(3):586–615, 2003.

[27] Victor S Miller. The weil pairing, and its efficient calculation. *Journal of Cryptology*, 17(4):235–261, 2004.

[28] Fangguo Zhang, Reihaneh Safavi-Naini, and Chih-Yin Lin. New proxy signature, proxy blind signature and proxy ring signature schemes from bilinear pairing. *IACR Cryptology ePrint Archive*, 2003:104, 2003.

[29] Steven D Galbraith. Supersingular curves in cryptography. In *Advances in Cryptology—ASIACRYPT 2001*, pages 495–513. 2001.

[30] Gerhard Frey and Hans-Georg Ruck. A remark concerning -divisibility and the discrete logarithm in the divisor class group of curves. *Mathematics of computation*, 62(206):865–874, 1994.

[31] Masahiro Mambo, Keisuke Usuda, and Eiji Okamoto. Proxy signatures: delegation of the power to sign messages. *IEICE transactions on fundamentals of electronics, communications and computer sciences*, 79(9):1338–1354, 1996.

[32] WD Lin and JK Jan. A security personal learning tools using a proxy blind signature scheme. In *Proceedings of International Conference on Chinese Language Computing*, pages 273–277. Illinois, USA, 2000.

[33] Dan Boneh, Ben Lynn, and Hovav Shacham. Short signatures from the weil pairing. In *Advances in Cryptology—ASIACRYPT 2001*, pages 514–532. Springer, 2001.

[34] Florian Hess. Efficient identity based signature schemes based on pairings. In *International Workshop on Selected Areas in Cryptography*, pages 310–324. Springer, 2002.

[35] Vedat Coskun, Busra Ozdenizci, and Kerem Ok. A survey on near field communication (nfc) technology. *Wireless personal communications*, 71(3):2259–2294, 2013.

[36] Yue-Shan Chang, Ching-Lung Chang, Yung-Shuan Hung, and Ching-Tsorng Tsai. Ncash: Nfc phone-enabled personalized context awareness smart-home environment. *Cybernetics and Systems: An International Journal*, 41(2):123–145, 2010.

[37] Jung Hee Cheon. Security analysis of the strong diffie-hellman problem. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 1–11. Springer, 2006.

[38] Surescript. `https://spring.io//`.

[39] Angelo De Caro and Vincenzo Iovino. jpbc: Java pairing based cryptography. In *Proceedings of the 16th IEEE Symposium on Computers and Communications, ISCC 2011*, pages 850–855, Kerkyra, Corfu, Greece, June 28 - July 1, 2011. IEEE.

[40] MS Windows NT kernel description. `http://www.mkyong.com/spring-security/spring-security-form-login-using-database/`. Accessed: 2010-09-30.

[41] Surescript. `http://hibernate.org//`.

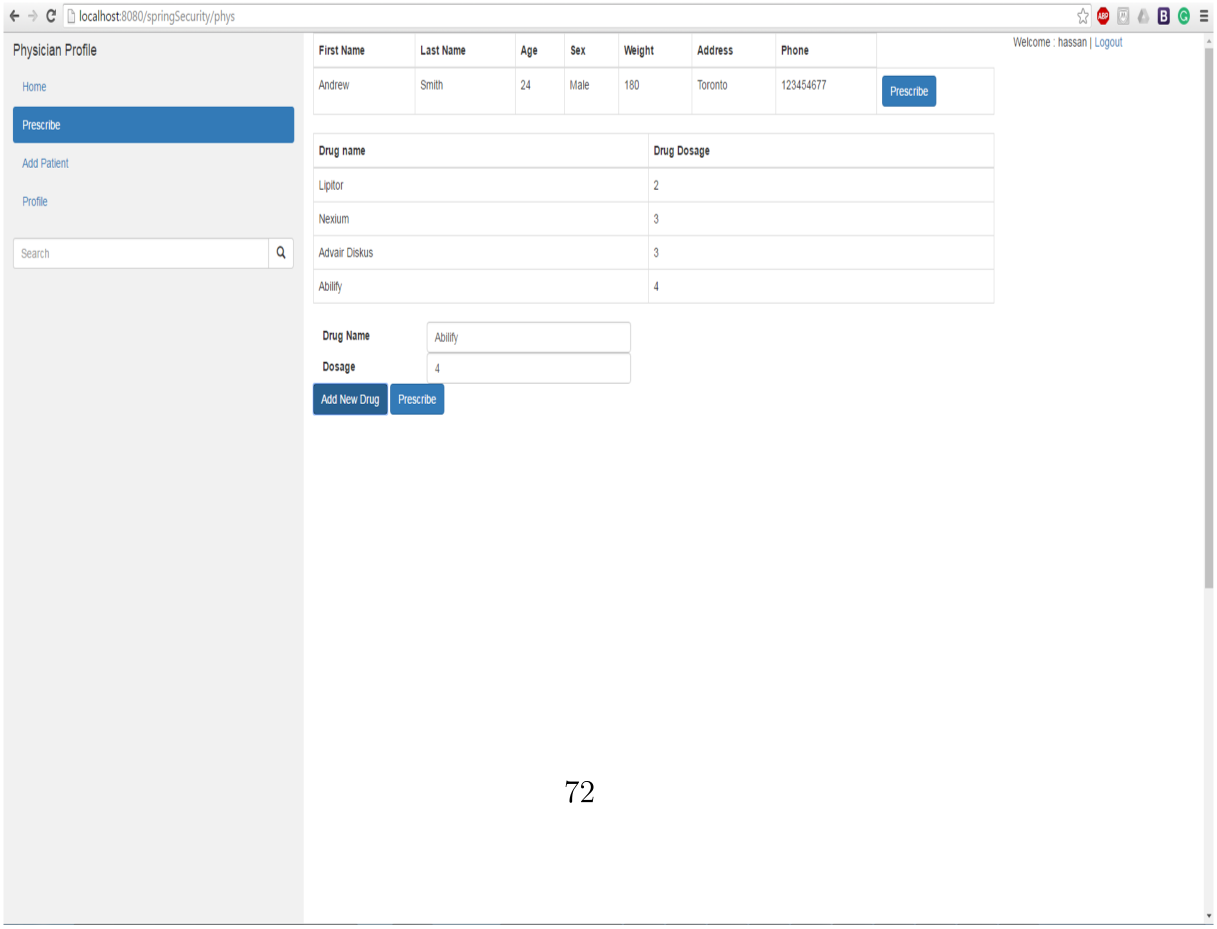
[42] Bradley Efron and Robert J Tibshirani. *An introduction to the bootstrap*. CRC press, 1994.

Appendix A

Appendix

A.1 Graphical User Interface

A.1.1 Web User Interface

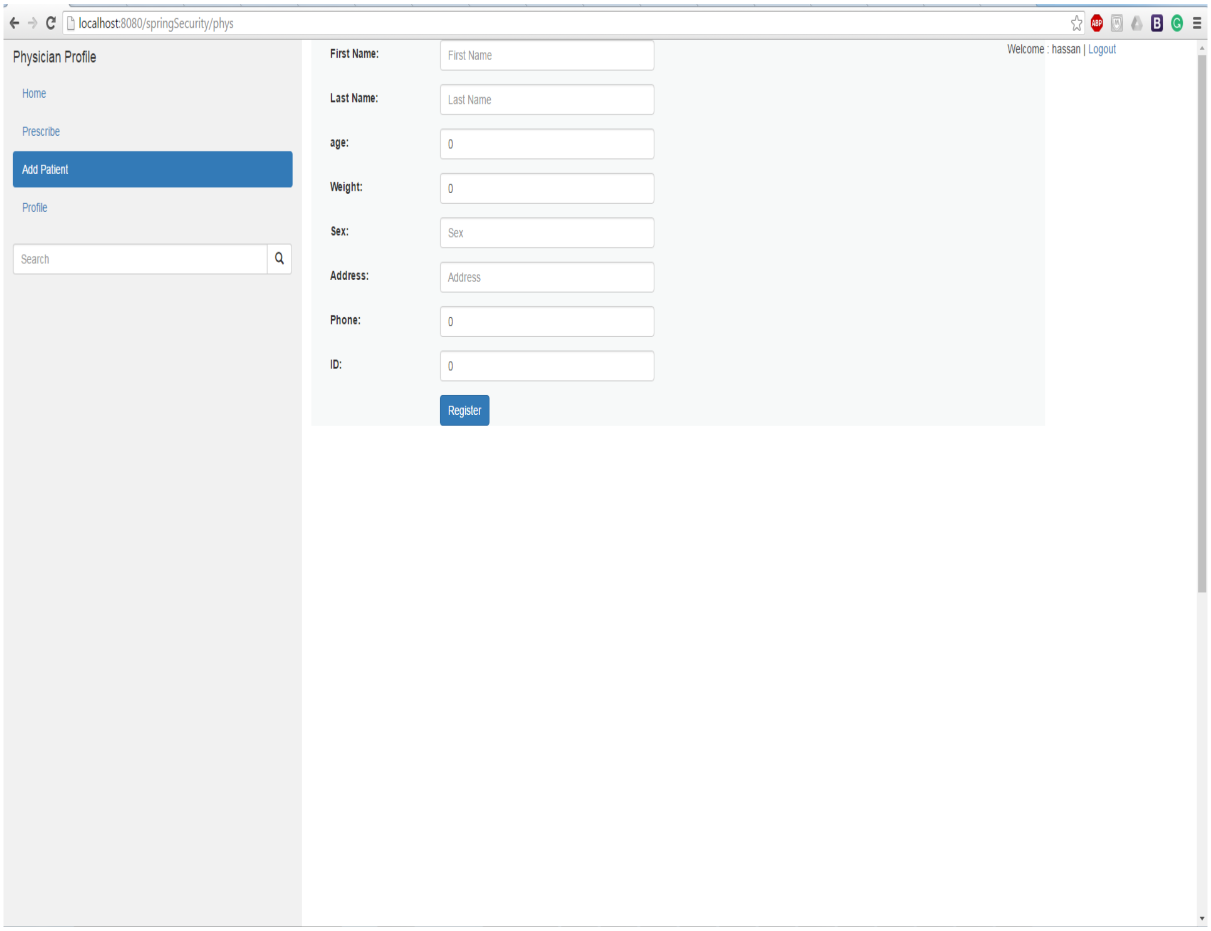


First Name	Last Name	Age	Sex	Weight	Address	Phone	
Andrew	Smith	24	Male	180	Toronto	123454677	Prescribe

Drug name	Drug Dosage
Lipitor	2
Nexium	3
Advair Diskus	3
Abilify	4

Drug Name: Abilify

Dosage: 4



localhost:8080/springSecurity/phys

Physician Profile

Welcome : hassan | Logout

First Name: First Name

Last Name: Last Name

age: 0

Weight: 0

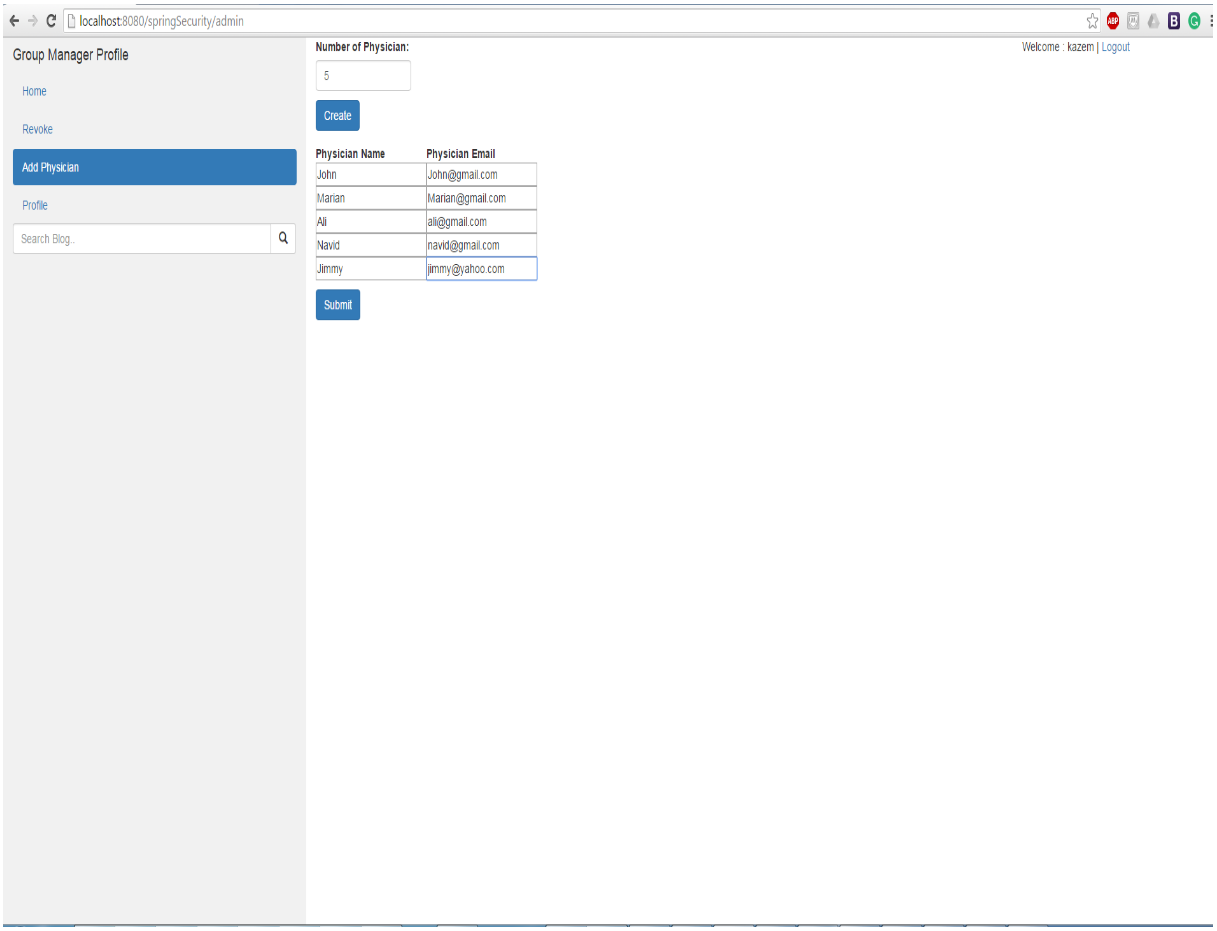
Sex: Sex

Address: Address

Phone: 0

ID: 0

Figure A.2: Add Patient tab on the Physician Panel.



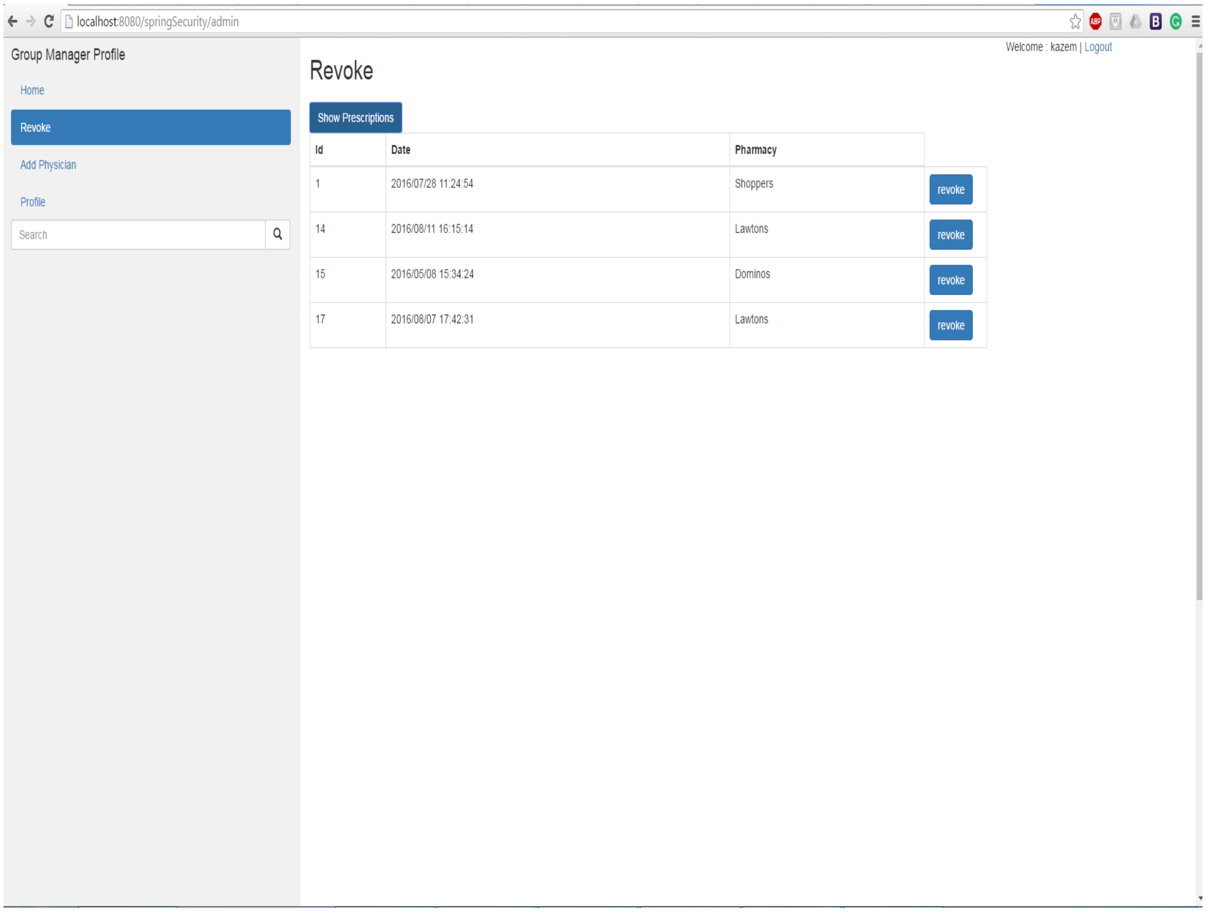
localhost:8080/springSecurity/admin

Welcome : kazem | Logout

Number of Physician: 5

Physician Name	Physician Email
John	John@gmail.com
Marian	Marian@gmail.com
Ali	ali@gmail.com
Navid	navid@gmail.com
Jimmy	jimmy@yahoo.com

Figure A.3: Add Physician tab on the Group Manager Panel.

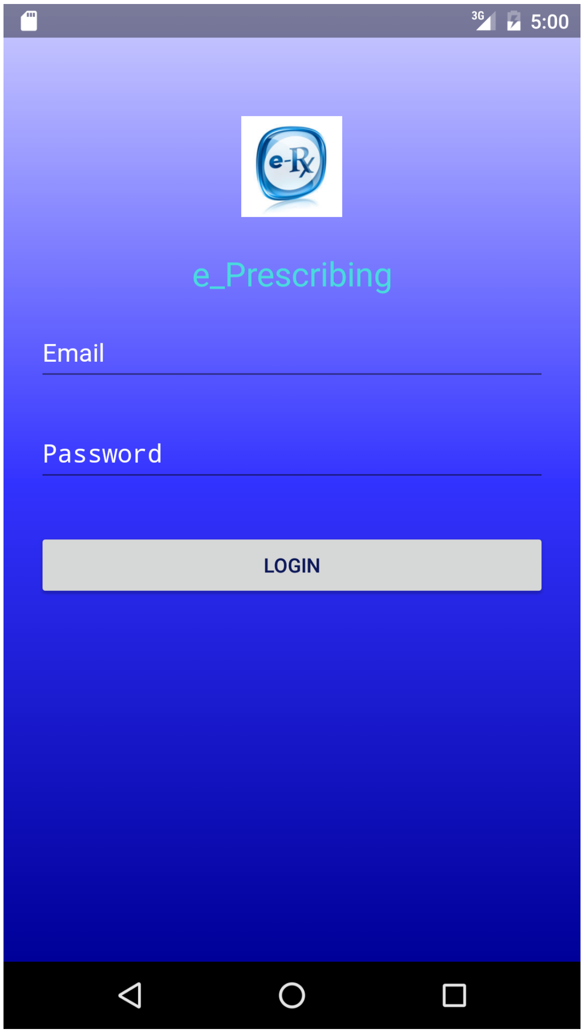


Revoke

Id	Date	Pharmacy	
1	2016/07/28 11:24:54	Shoppers	revoke
14	2016/08/11 16:15:14	Lawtons	revoke
15	2016/05/08 15:34:24	Dominos	revoke
17	2016/08/07 17:42:31	Lawtons	revoke

Figure A.4: Revoke page of the group manager.

A.1.2 Mobile User Interface


5:00

e_Prescribing

Email

Password

LOGIN

Figure A.5: Login page on mobile application.

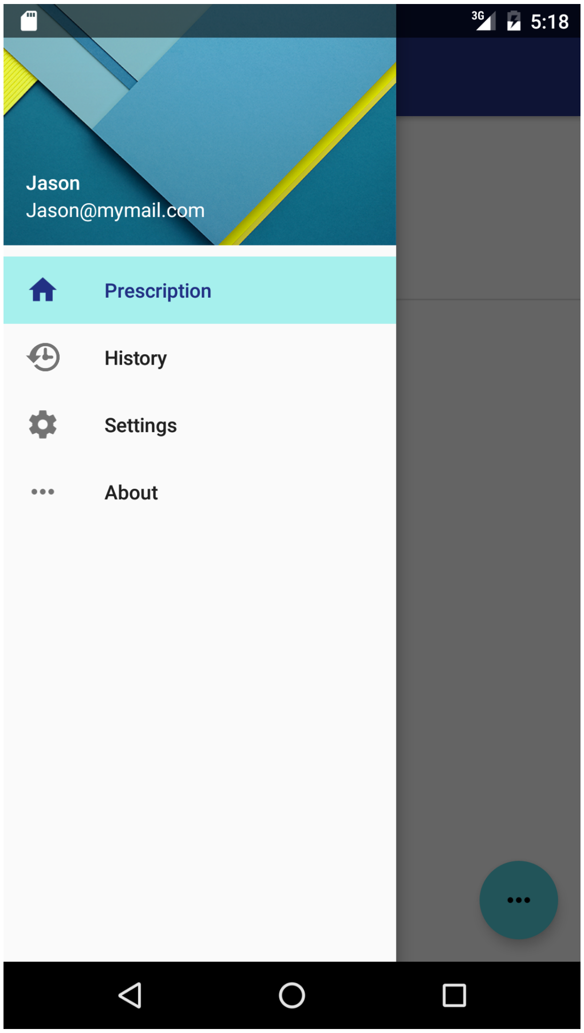
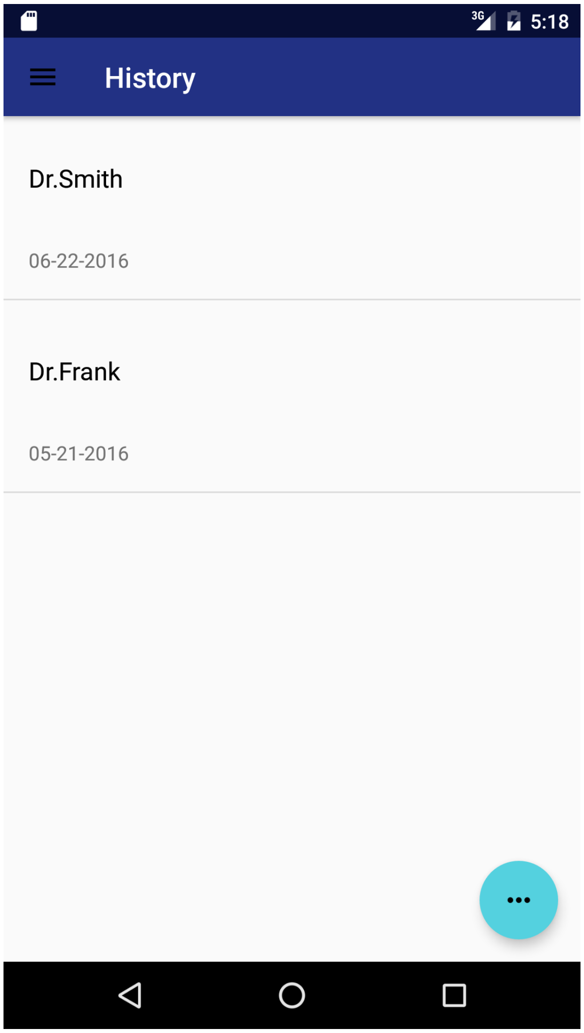


Figure A.6: Home page on mobile application.



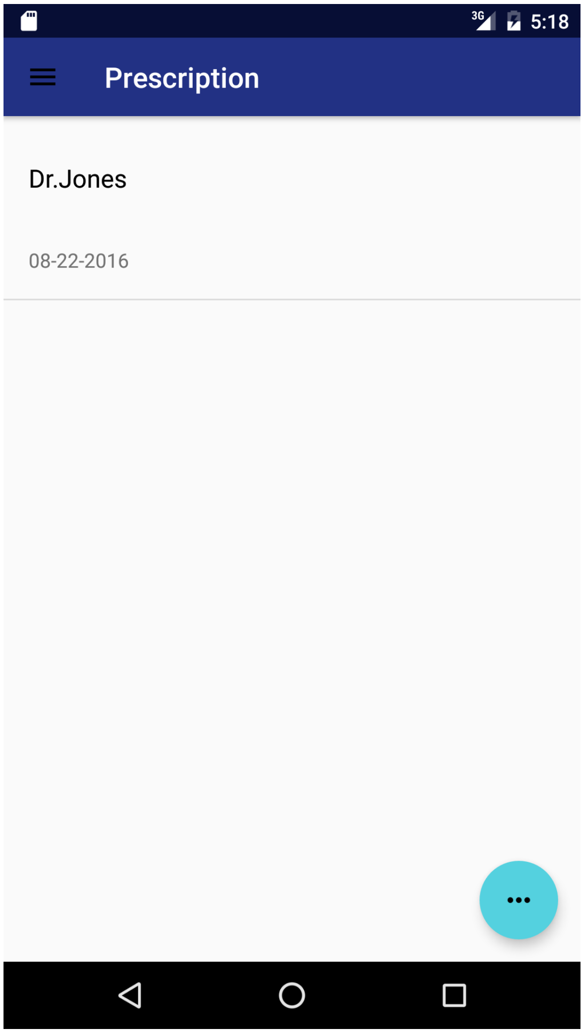
Dr.Smith

06-22-2016

Dr.Frank

05-21-2016

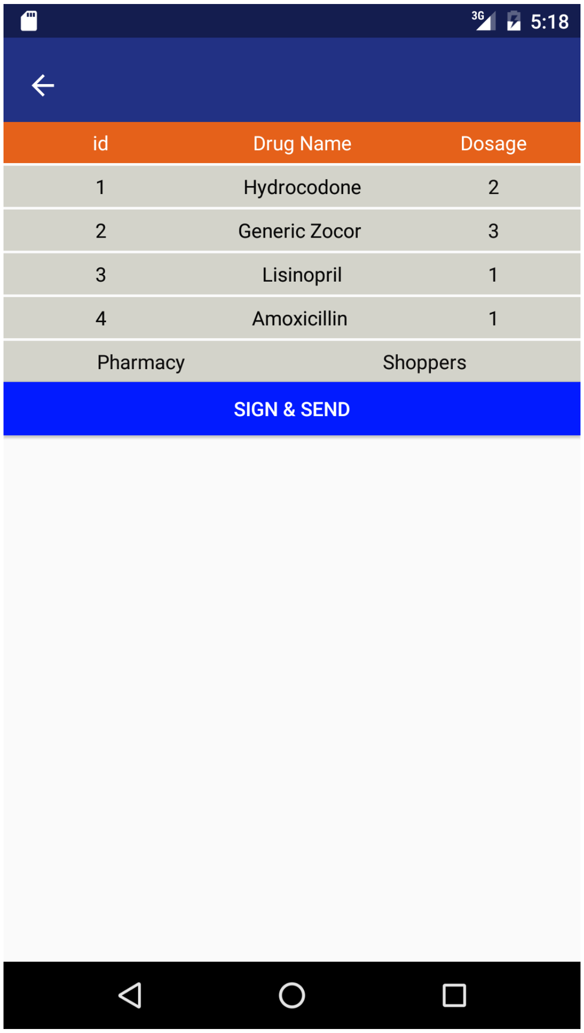
Figure A.7: New prescription on mobile application.



Dr.Jones

08-22-2016

Figure A.8: History page on mobile application.

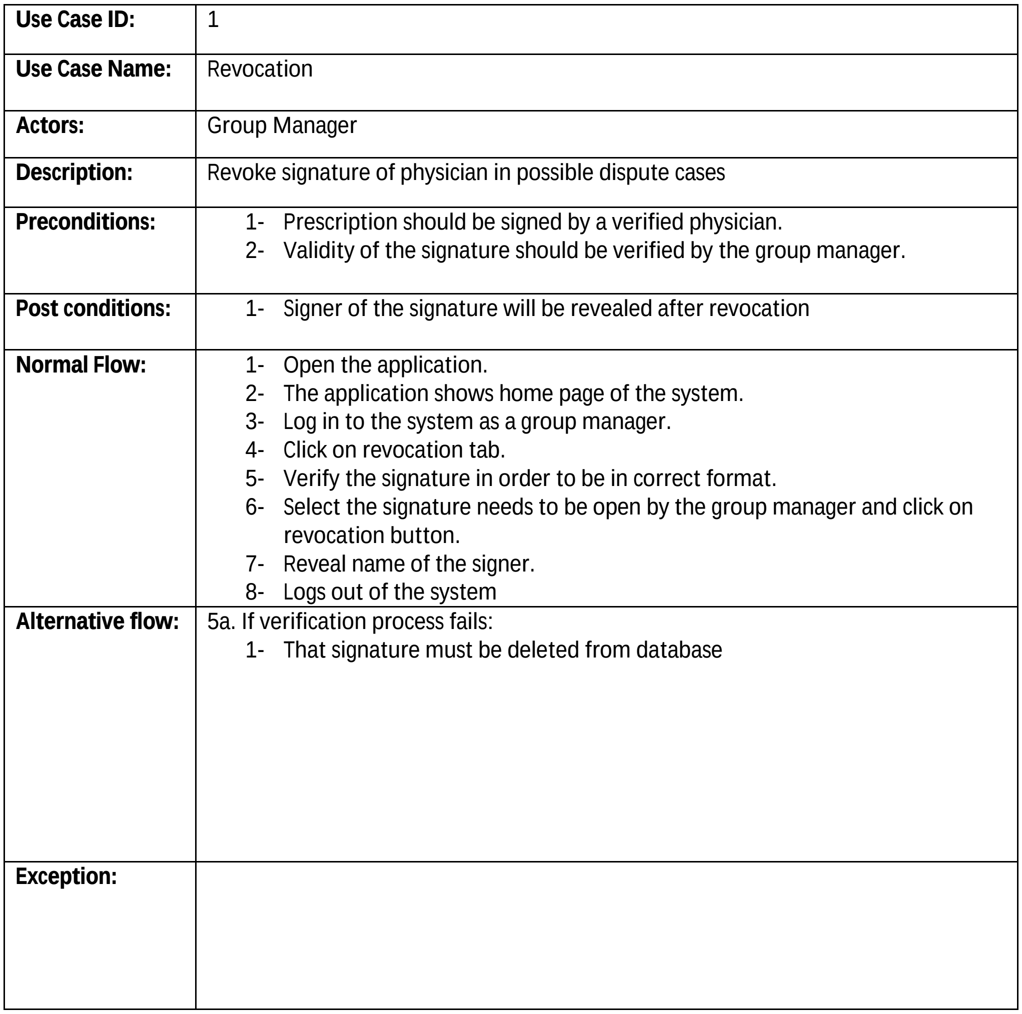


id	Drug Name	Dosage
1	Hydrocodone	2
2	Generic Zocor	3
3	Lisinopril	1
4	Amoxicillin	1

Pharmacy	Shoppers

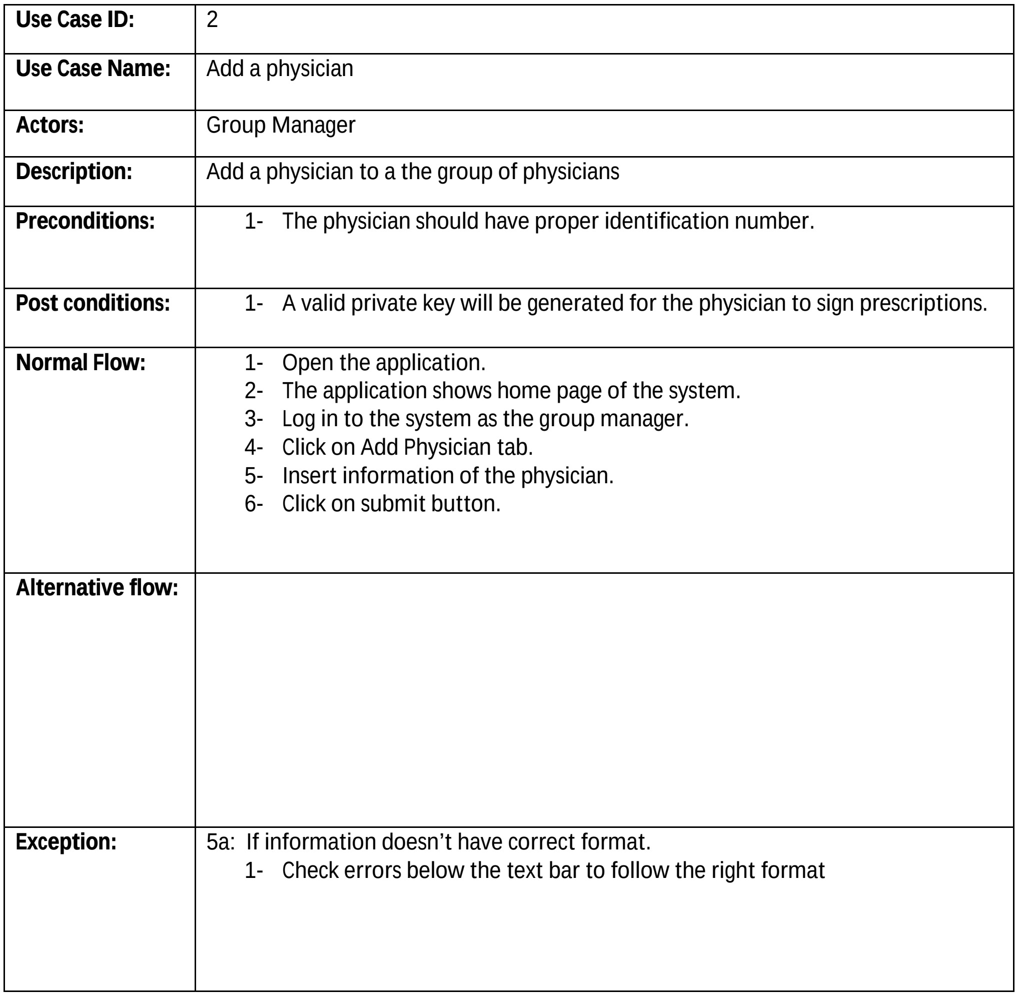
Figure A.9: Prescription content.

A.2 Use Case Description Diagrams



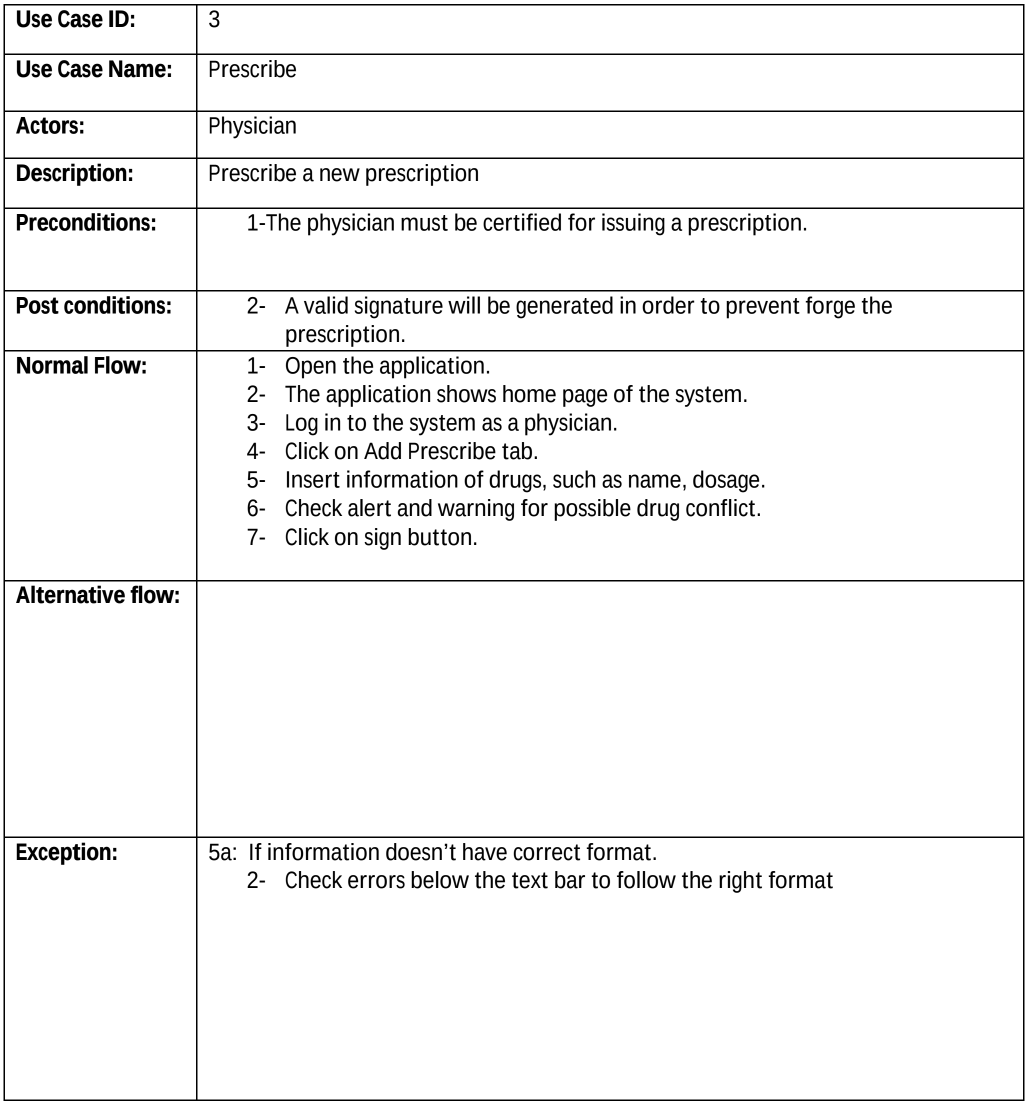
Use Case ID:	1
Use Case Name:	Revocation
Actors:	Group Manager
Description:	Revoke signature of physician in possible dispute cases
Preconditions:	1- Prescription should be signed by a verified physician. 2- Validity of the signature should be verified by the group manager.
Post conditions:	1- Signer of the signature will be revealed after revocation
Normal Flow:	1- Open the application. 2- The application shows home page of the system. 3- Log in to the system as a group manager. 4- Click on revocation tab. 5- Verify the signature in order to be in correct format. 6- Select the signature needs to be open by the group manager and click on revocation button. 7- Reveal name of the signer. 8- Logs out of the system
Alternative flow:	5a. If verification process fails: 1- That signature must be deleted from database
Exception:	

Figure A.10: Revocation Use Case.



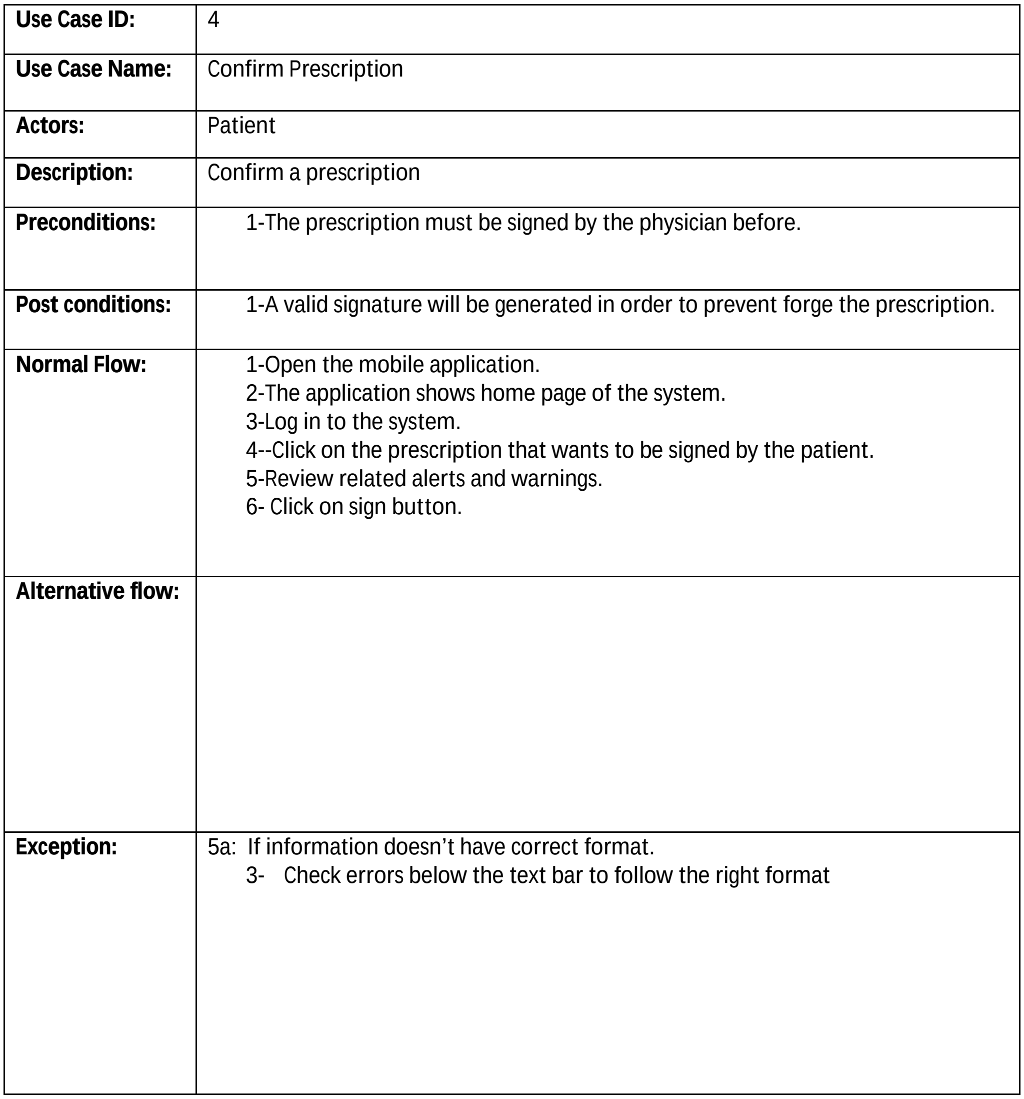
Use Case ID:	2
Use Case Name:	Add a physician
Actors:	Group Manager
Description:	Add a physician to a the group of physicians
Preconditions:	1- The physician should have proper identification number.
Post conditions:	1- A valid private key will be generated for the physician to sign prescriptions.
Normal Flow:	1- Open the application. 2- The application shows home page of the system. 3- Log in to the system as the group manager. 4- Click on Add Physician tab. 5- Insert information of the physician. 6- Click on submit button.
Alternative flow:	
Exception:	5a: If information doesn't have correct format. 1- Check errors below the text bar to follow the right format

Figure A.11: Add a Physician Use Case.



Use Case ID:	3
Use Case Name:	Prescribe
Actors:	Physician
Description:	Prescribe a new prescription
Preconditions:	1-The physician must be certified for issuing a prescription.
Post conditions:	2- A valid signature will be generated in order to prevent forge the prescription.
Normal Flow:	1- Open the application. 2- The application shows home page of the system. 3- Log in to the system as a physician. 4- Click on Add Prescribe tab. 5- Insert information of drugs, such as name, dosage. 6- Check alert and warning for possible drug conflict. 7- Click on sign button.
Alternative flow:	
Exception:	5a: If information doesn't have correct format. 2- Check errors below the text bar to follow the right format

Figure A.12: Prescribe Use Case.



Use Case ID:	4
Use Case Name:	Confirm Prescription
Actors:	Patient
Description:	Confirm a prescription
Preconditions:	1-The prescription must be signed by the physician before.
Post conditions:	1-A valid signature will be generated in order to prevent forge the prescription.
Normal Flow:	1-Open the mobile application. 2-The application shows home page of the system. 3-Log in to the system. 4--Click on the prescription that wants to be signed by the patient. 5-Review related alerts and warnings. 6- Click on sign button.
Alternative flow:	
Exception:	5a: If information doesn't have correct format. 3- Check errors below the text bar to follow the right format

Figure A.13: Confirm Prescription Use Case.