UNIVERSITY OF NAIROBI

SCHOOL OF COMPUTING AND INFORMATICS

A RECOMMENDER PRIVACY AWARE REPUTATION-BASED HEALTHCARE SYSTEM

BY

LINCOLN KASENGI KITHANDI

P54/79266/2015

SUPERVISOR

DR. DANIEL ORWA

RESEARCH PROJECT SUBMISSION IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT.

NOVEMBER 2016

DECLARATION

This project as presented in this report is my original work and has not been presented for any other institutional award.

Sign: ________________________________ Date: ___________________________

Kithandi Lincoln Kasengi

P54/79266/2015.

This project has been submitted in partial fulfilment of the requirements for the degree of Masters of Science in Information Technology Management at the University of Nairobi with my approval as the university supervisor.

Sign: ______________________________________ Date: _____________________

Dr. Daniel Orwa

School of Computing and Informatics

University of Nairobi.

ACKNOWLEDGEMENT

I would like to acknowledge the treasured support I received from my parents; Fredrick K. Mbindyo and Loice Mwau, my siblings; Mwende, Mbenya, Katua, Beatrice and Dennis. Your support and encouragement will forever be remembered.

I wish to express my sincere gratitude to my supervisor, Dr. Daniel Orwa for his tireless assistance, guidance and support throughout the research process. I am grateful to him for his availability during the whole process.

I would also like to extend my heartfelt gratitude to my lecturers, classmates and friends, thank you for all your support and prayers.

Above all, I thank the Almighty God for guiding me throughout my academic life and letting me reach this far.

ABSTRACT

Over the last decade, recommender systems have become an increasingly researched area due to the increased usage of the Internet by users. These systems have been widely applied in e-commerce and service oriented networks. In this study, a new approach has been proposed for the development of a recommender system which deals in a new domain, i.e., healthcare, giving a different edge to these sorts of systems altogether. In this paper, it has been demonstrated how a recommender system can be used in the healthcare domain to provide recommendation to interested users about the optimal healthcare providers. Moreover, given that patient information is sensitive data and need to be protected, we propose a recommender system that takes sensitivity of healthcare information into account. In our proposed architecture, a healthcare user submits ratings anonymously to the system and thus the identity of the user is concealed. Moreover, a user who intends to obtain a recommendation from the system can obtain such recommendation without compromising his identity. Either way, the identity of any user interacting with the system is preserved. Our proposed architecture is reputation-based and allows for the computation of real-valued rankings of healthcare service providers hence giving more fine-grained recommendations.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

PPDM	Privacy Preservation Data Mining
SMC	Secure Multiparty Computation
TA	Trusted Authority
CA	Certification Authority
HU	Healthcare user
HSP	Healthcare service provider
DVD	Digital Video Disk
PCC	Pearson Correlation Coefficient
API	Application Programming Interface
PHP	Hypertext Preprocessor

CHAPTER ONE: INTRODUCTION

1.1. Background

It is natural that when we seek healthcare services we want the best service(s). Consequently, finding appropriate healthcare providers to diagnose and treat health conditions is one of the vital decisions that a patient must make. Oftenly, patients rely on friends and family for advice on where to seek treatment. However, recommendations from friends may not be reliable as they may not have experience with the same medical history as the patient requesting recommendation. Moreover, it may be difficult for a patient in a new locality to receive recommendations as the patient doesn't have a trusted network from which to seek advice. In some other cases, patients can obtain healthcare recommendations from public information available on, e.g. the internet. However, such information may not be readily available as medical records and other patient information is sensitive and confidential information.

More often than not, the only sure way to provide accurate recommendations is through the help of recommender systems. A Recommender s2 is a personalized service system that can assist the user in making decision by filtering the information according to his need and interest and then recommend him appropriate items, thus helping the user in finding the preferred items. Recommendation systems are used in a variety of domains like recommending web pages, restaurants, television programs, movies, music and items for sale. Many algorithms have been developed in order to provide recommendations; however, it is a possibility that an algorithm might work well for certain kinds of recommendations but might perform poorly and degrade the performance for some. In this study, a new approach has been proposed for the development of a recommender system which deals in a new domain, i.e., healthcare, giving altogether a different edge to these sorts of systems. In this paper, it has been demonstrated how a recommender system can be used in the healthcare domain to provide recommendation to interested users about the optimal healthcare providers. Moreover, given that patient information is sensitive data ad need to be protected, we propose a recommender system that takes sensitivity of healthcare information into account. In our proposed architecture, a healthcare user submits ratings anonymously to the system and thus the identity of the user is concealed. Moreover, a user who intends to obtain a recommendation from the system is able to obtain such recommendation without compromising his identity. Either way, the identity of any user interacting with the system is preserved. Our proposed architecture is reputation-based and allows for the computation of real-valued rankings of healthcare service providers hence giving more fine grained recommendations.

1.2. Problem Statement

Given that technological innovation is the major driver of sustainable economic growth and impacts (in both a positive and negative manner) across most aspects of human society, it could be argued that this is reason enough for any research into the phenomenon. However, the call for broad research in how we can improve healthcare is also an important rationale, the overall premise being the belief that a healthy nation is a wealthy nation. Medical services are critical in nature and when a patient develops a new condition, they want to obtain medical services from a reputable and relevant medical service provider. Despite the zeal to get the best services from a medical service provider, many times patients get unsatisfactory healthcare services. In some cases the poor services may lead to death or certain health complications. Moreover, some medical service providers have persistently offered poor services yet they still get clients due to a lack of way to determine the quality of services beforehand. There is therefore need for a way to determine which medical providers offer quality services to medical users before they engage in any service with the provider. More often than not, the only sure way to determine whether services offered by a service provider are good or not depends on recommender systems. Reputation-based systems have been widely used in fields such as ecommerce. However, the same is largely lacking in the medical field. There is considearabley little research on reputation systems in healthcare, despite the critical nature of healthcare services as opposed to other kinds of service provision such as ecommerce. In this study, we propose a privacy aware reputation-based recommender healthcare system that will enable healthcare users get recommendations for the optimal healthcare provider for specific health conditions. The healthcare user will be able to know beforehand the reputation score of a certain healthcare provider prior to obtaining any service from the provider. This way, the system will help discourage interactions with parties that continually offer poor services to patients and still attract patients who seek good services. A patient seeking for any service from the healthcare service providers will have an opportunity to choose the one with the best reputation from the providers who offer similar services.

1.3. Objectives

The following objectives will guide this study;

1. To design an architecture for a recommender reputation based healthcare system for use in providing healthcare provider recommendations to healthcare users.
2. To implement a privacy aware reputation-based health recommender system for use in providing healthcare recommendations to interested healthcare users.
3. To use a set of test cases to test the developed healthcare recommender system to determine whether it works correctly.

1.4. Significance of the Study

At the end of this research, we aim to develop a privacy-preserving reputation based healthcare system that will help medical users get recommendations of the best ranked medical service providers and hence can access quality services.

There has been considerably little research on health recommender systems. This research, therefore, aims to add to the knowledge of this area and will be useful to researchers and academicians who may find this work relevant for their use in further expounding their research.

Additively, recommending a patient to obtain service from a particular service provider will benefit the service providers themselves, and therefore, this research will be useful to medical service providers who will be able to get a reputation score from patients.

1.5. Assumptions and Limitations of the Study

For our proposed architecture to be realized and implemented, several assumptions are made. First, we assume that the system maintains a list of healthcare providers and health conditions for which recommendations can be provided. This way, healthcare users will be able to submit ratings from the list of healthcare service providers provided based on specific health conditions. Moreover, we assume that a rating criterion (rating specification) for a medical provider already exists. It is worth noting that ratings can be the result of a broad range of questions such as overall satisfaction, time until cured, which are outside the scope of this work. Furthermore, we assume that the user rankings are numeric in nature. This enables us to assume that an "average" rating makes sense, and is consistent across the recommendation system. While there are some known challenges with recommender systems such as ballot stuffing and shilling attacks (where a healthcare provider attempts to sabotage a competitor to make themselves better), we recommended that the techniques that already exist to combat these attacks be extended for their systems. Finally, in the rest of this work we assume that a recommendation is given for a specific health condition and is computed from ratings submitted by patients.

CHAPTER TWO: LITERATURE REVIEW

2.0. Introduction

Significant research has been conducted in the area of privacy preserving recommendation systems. In this chapter, we shall discuss the existing literature on recommender systems followed by a description of privacy preservation techniques and finally we look at challenges facing most of the proposed recommender systems.

2.1. Recommender Systems

Recommender systems have stirred up a lot of research interest over the last decade as they enable personalized recommendations and services to users. These systems which explore user behavior and user ratings to improve the recommendation process rely heavily on the amount collected from the users. This information is mostly privacy-sensitive and open to being abused by the service provider himself if not protected properly. As such, there has been equally increased research into recommendation systems that are privacy preserving.

Recommender systems have been widely categorized into five basic techniques: Content-based, collaborative, demographic, utility-based and knowledge-based (Burke, 2007).

Jeckmans et al. (2013) also follow Burke (2007) and consider collaborative, content-based, demographic, and knowledge-based filtering approaches as the basic recommender types.

Burke (2007) states that all recommender systems employ at least one of these basic techniques that “have complementary advantages and disadvantages.” The final category of recommender system algorithms is, in fact, hybrid recommender systems, that combine multiple techniques to achieve synergy and avoid the weaknesses that each type individually has.

2.1.1. Collaborative Filtering (CF)

Collaborative filtering system is a system that helps people make choices based on the opinions of other like-minded people (Resnick, Iacovou, Suchak, Bergstorm & Riedl, 1994).

The earliest recommender systems used collaborative filtering systems and were generally designed to give users information about items. Overtime, collaborative filtering systems have been enhanced and used to personalize user content (Schafer et al., 2007). Collaborative filtering has at present gained a lot of research interest from scholars due to the increased usage of ecommerce and the availability of electronic word-of-mouth (eWOM) recommendations. Through electronic word-of-mouth, it has been possible to use thousands or even millions of opinions from a community such as a social media site to get "a truly personalized view" on an item (Schafer et al., 2007).

Collaborative filtering approaches are typically divided into *user-based* and *item-based approaches* (Schafer et al., 2007; Cremonesi et al., 2008; Konstan & Riedl, 2012; Jeckmans et al., 2013).

User-based collaborative filtering approach is based on finding the most similar users, referred to as neighbors, for the current user. Therefore recommendations are made for the top N items after a rating aggregation has been done, based on the neighbor ratings (Konstan & Riedl, 2012). Several metrics have been used to compute the similarity between users in order to find the *k* nearest neighbors. The most commonly used traditional metrics include *Pearson correlation*, *cosine*, *adjusted cosine*, *constrained correlation*, *mean squared difference*, and *Euclidean* (Bobadilla et al., 2013).

Item-based CF approaches, on the other hand, use items instead of users to give recommendations. In other words, item-based CF approach generates recommendations by finding similar items to that the current user has (Schafer et al., 2007).

Generally, Collaborative filtering techniques are heavily reliant on wisdom of the crowd and generate recommendations to a certain user based on neighboring user preferences.

Whereas CF approaches have been widely used to generate recommendation, they are faced with several shortcomings. First is the data sparsity problem: This is mainly due to having too many items in the system but there exists few items shared among users. Second is the cold-start problem: This is whereby a user or item has a small number of ratings and hence it will be difficult to find an accurate neighborhood. Third is the scalability problem: This is whereby CF approach may encounter severe performance and scaling issues especially when the number of users and items increase (Abdullah, 2012).

2.1.2. Content-Based Filtering (CBF)

Content-based filtering (CBF) is an item-to-item correlation system whereby the recommendations are based on attributes associated with items and preference data, e.g. ratings that users have given to items (Burke, 2007). While collaborative filtering is based on the assumption that people with similar tastes rate things similarly, content based filtering is based on the assumption that items with similar objective features, or attributes, are rated similarly (Schafer et al., 2007; Bobadilla et al., 2013). This system works well with items that are described by text, such as news articles, research papers and books. In other words, items are recommended based on information about the item itself rather than on the preferences of other users (Zhou et al., 2012). In effect, content-based recommenders suggest to a user items the content of which is similar to the content of the items that the user has rated positively, or has otherwise shown preference for, in the past (Bobadilla et al., 2013). In simple terms, a content-based approach learns a profile of the user's interests based on the attributes present in the items that the user has rated positively (Burke, 2002).

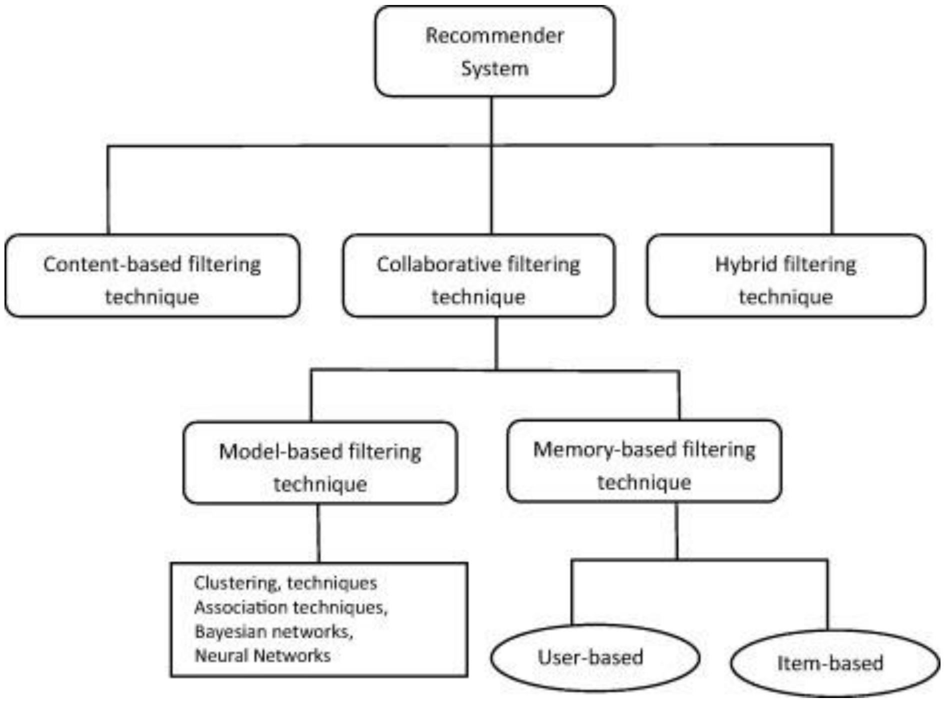


Figure 1: Recommendation techniques.

2.1.3. Hybrid Recommendation approaches

Hybrid recommendation approaches employ a mixture of two or more recommendation techniques so as to achieve synergy and avoid the weaknesses that each type individually has. Currently, hybrid Recommender systems have been utilized in order to combine the advantages of various recommendation approaches (Hussein et al., 2014). Hybrid recommender systems, however, consume a lot of resources and incur heavy computational load (Burke, 2002; Liang, 2010).

2.1.4. Demographic recommender systems

Demographic recommender systems recommend items based on the demographic characteristics of users. An example of demographic recommender at work could be the display of ads to users depending on the country they are accessing the system or the language they are speaking.

2.1.5. Knowledge-based recommender system

Knowledge-baed recommender systems recommend items to users based on specific domain knowledge on how particular item features meet users' needs and preferences and, ultimately how the item is useful for the user. In such systems, a similarity function estimates how much the user needs (problem description) match the recommendations (solutions of the problem).

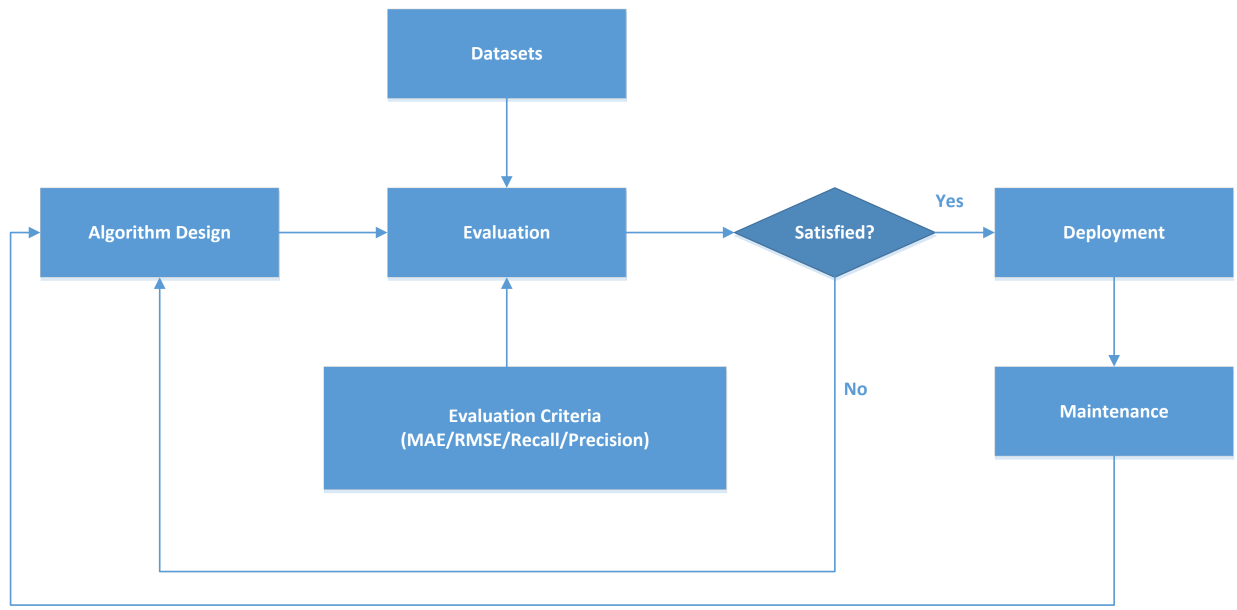


Figure 2: Flowchart for the design of a recommender system

Source: Zhang, F et al (2016), Fast algorithms to evaluate collaborative filtering recommender systems.

2.2. Reputation Systems

People are increasingly dependent on information online to decide whether to trust a particular object or not. Reputation systems are therefore an essential part of any e-commerce or product review website, where they provide methods for collecting and aggregating users' ratings to calculate the overall reputation scores for products, users, or services (Resnick et al., 2000). The existence of reputation scores in these websites helps people making decisions about whether to buy a product or to use a service or not. Reputation systems play a significant role in users' decision-making process. Reputation systems consist of three major components, as we illustrate in *Figure 3* (Jøsang et al., 2007). The first element is the feedback collection from users. In this stage, reputation systems describe the methods used for collecting users' feedback; that is, centralized or distributed. They also describe what sort of feedback to be gathered, such as user ratings, textual reviews, or critics' and experts' reviews. This stage may involve opinion mining techniques to detect opinion polarity and strength in textual reviews and then represent them as numerical scores (Abdel-Hafez & Xu, 2013a). The output of the feedback collection stage is a set of ratings towards items to be used in the reputation engine for generating reputation scores.

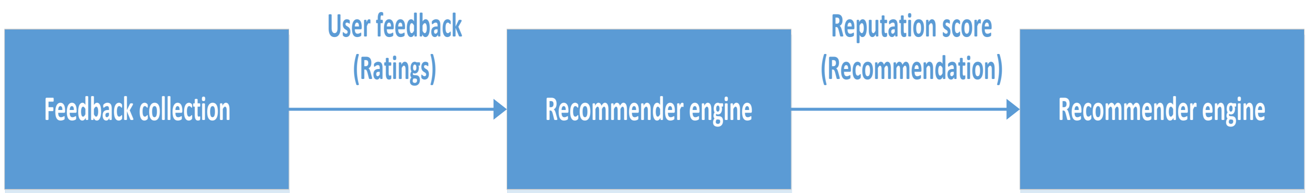


Figure 3: Reputation systems components

Source: Josang et al. (2007), An accurate rating aggregation method for generating item reputation.

2.3. Reputation-Based Recommender Systems

Reputation-based systems have recently become an interesting area of research. Researchers have put much emphasis on how to improve accuracy and reliability of recommender systems by combining recommender systems with reputation systems. In sectors that deal with service delivery like healthcare, reputation and trust issues are very vital.

Reputation systems are employed to provide users with advice on the quality of items on the web, based on the aggregated value of user-based ratings. Recommender systems are used online to suggest items to users according to each user's expressed preferences. Yet recommender systems will endorse an item regardless of its reputation value. In our proposed work, we introduce novel methods to combine recommender and reputation systems in order to enhance the accuracy of the top-n recommender results (Abdel-Hafez, Tang, Tian, & Xu, 2014).

In a recent study, Ku & Tai (2013) proposes an exploratory framework to investigate the effect of recommendation systems and reputation systems on purchase intentions regarding recommended products from an information communication perspective. Their experiment included 48 participants, who were offered a discount to a movie DVD e-store. They collected data about their preferences to generate recommendations and then asked all of the participants to complete a questionnaire about each recommended movie. Their results show that the relevance between users' preferences and recommended items intensifies consumer attitudes towards the purchase of the recommended product. Moreover the opinions of other consumers influence consumer attitudes towards the purchase of the recommended product via normative social influence, which requires that recommendation systems should also consider online review to increase their persuasiveness to consumers. The recommender system recommends a list of items that reflects the opinions of a local community of similar users, with these recommendations personalized for each user. In contrast, the reputation system provides the opinions of the whole community. Jøsang, Pini, Santini, and Xu (2013) suggest that combining reputation scores with recommendation scores will provide more accurate recommendations.

They use a CF method to recommend the top-K most similar items, where finding nearest neighbors depends on the PCC similarity function. On the other hand, they use the belief model they introduced in a previous work (Jøsang, 2001) in order to calculate reputation scores. The authors mention different methods for combining resulted scores, but they adopt the cascading minimum common belief fusion (CasMin) method. This method ensures that the values from both systems, recommender and reputation, must be high in order to produce a high value in the CasMin method. However, there was no experiment to prove that the recommendations created using their method are better.

It is worth noting that most of the existing reputation-based recommender systems do not consider the distribution of ratings (Hu et al., 2009). People usually have different levels of leniency when rating an item, depending on their preferences and expectations. For example, lenient users would rate an item as 5 stars if they have a minor negative opinion about it, while strict users would rate an item as 4 stars because they are harder to satisfy. We believe that the reputation system must acknowledge that both ratings are positive ones. Given the previous example, if we use the rating scale $[1-5]$, then the rating levels of 4 and 5 indicate positive opinions, 1 and 2, indicate negative opinions, and 3 indicates neutral opinions. The distributions of positive and negative ratings for an item should influence its reputation. Looking at a simple example, if we have an item with 7 ratings {2,2,2,2,3,5,5}, we can say that we have 4 negative, 1 neutral, and 2 positive opinions. Because of the high frequency of rating level 2, rationally, the reputation for this item should be less than 3. However, the mean of the ratings is 3.0, which is considered neutral. In other words, the overall reputation score of a specific product can be skewed towards the negative, even when the number of positive ratings is higher than the negative ones if the count of ratings is not taken into consideration, and vice versa.

As mentioned previously, the weighted average is currently the most used method for ratings aggregation, while the weights usually represent the time when the rating was given, or the reviewer reputation. In the simplest case, where we don't consider other factors, the weight for each rating is $1/n$, if there are n ratings to an item (this is the naive method). No matter that the simplest average method or the weighted-average methods take time or other user-related factors into consideration, the frequency of each rating level is not explicitly considered. Considering the ratings example, for the simplest average method, the weight for each of the ratings is even though the rating level 2 has a higher frequency than the other two rating levels. For other weighted-average methods, the weights are only related to time or user-related factors but not rating frequency.

The proposed recommender reputation-based healthcare system uses the weighting methodology. The recommender system uses weighted mean as a ratings aggregator, where the weighting factors include healthcare provider attributes. This way, the recommender healthcare system makes recommendations of the optimal healthcare providers based on aggregated patients' ratings.

2.4. Privacy Preservation Techniques

Privacy concerns in recommendation systems have been raised by various authors. Various privacy-preservation techniques have thus been proposed by the researchers. Basically, these privacy preserving techniques were derived from the already existing data hiding techniques. In data hiding techniques, data is altered, blocked, or trimmed out from the original database to avoid it being compromised by unauthorized persons (Polat and Du, 2005). *Figure 4* shows the different approaches used for hiding the data.

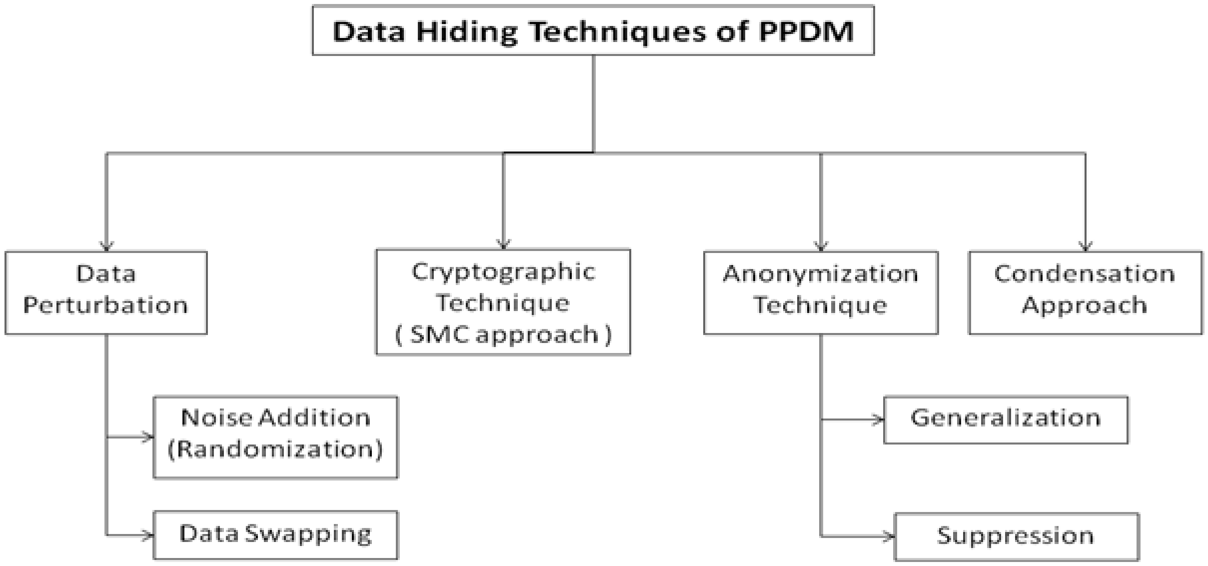


Figure 4: Data Hiding Techniques of PPDM

Source: Suchistra (2015), Techniques for Privacy Preservation in Data Mining

In Data Perturbation technique, available data is modified before it is passed to Data Mining. There are some ways to amend the data like swapping and adding noise, but after modification quality of the released data is maintained (Manish & Chaudhary, 2013).

Apart from data pertubation, privacy can be achieved by using cryptographic and secure multiparty computation (SMC) techniques. Cryptographic techniques are often utilized in distributed data mining. These cryptographic techniques have been borrowed to preserve privacy in recommender systems. Mbandu & Kamenyi (2015) use a variant of elgamal encryption to encrypt medical user information. A medical user who submits a reputation request system is presented with a public key which will be used for encryptionand decryption.

Another technique that has been employed in preserving privacy is anonymization technique. In this techniques, User identifying information is removed from the original data hence protecting the identity of the user. Many anonymization methods have been proposed in literature. One popular technique uses the k-anonymization approach.

In addition to these data hiding techniques, another technique is the condensation technique. Here, raw input data is compressed and packed into multiple groups referre to as clusters. The statistics of data in each cluster is then analyzed and maintained separately for each group. The statistics is then used to generate pseudo data for corresponding values. The user then publishes this pseudo data instead of the original data thus ensuring privacy as original data remains hidden from other parties(Gayatri Nayak, 2011).

2.5. Challenges of Recommender Systems

Recommender systems have been faced with several challenges some of which are as follows:

2.5.1. Unfair ratings

Sometimes users involved in providing ratings may submit ratings that are unfairly positive or unfairly negative. Moreover, some users may tend to give higher ratings or lower ratings than others because they may have some leniency when providing ratings. There are a number of ways in which this problem can be dealt with. Abdel-Hafez (2016) proposes a latency-aware quality (LQ) model which emphasizes that a user's rating tendency is used as weight. This way, healthcare users will be classified into lenient or strict users, and then use the leniency value, which is classified as a weight for each user's ratings. Another technique which has been used for normalization is the z-score method.

2.5.2. Change of identities

Sometimes healthcare service providers and other parties that have suffere significant loss of reputation can decide to change their identity and use a different name. This way, they'll be able to de-link themselves from the past and start afresh. This challenge can be dealt with by discouraging a change of identities through penalizing newcomers (Burke et al, 2006).

2.5.3. Low incentive for providing rating

User ratings are normally provided after a transaction has taken place hence the users may have no direct incentive for providing ratings about the accessed service. Resnick & Zeckhauser (2002) found out that 60.7% of buyers and 51.7% of sellers on eBay provided ratings. By ensuring patients of privacy of their ratings, the patients may be much willing to provide a rating of a medical provider to the system.

2.5.4. Bad mouthing and Boosting (Ballot stuffing)

Bad mouthing is said to occur when a patient (potentially offended) attempts to lower the score of a healthcare provider (Burke et al, 2006). Boosting is said to happen if, instead of lowering a score, the patients collude to increase a rating. In e-commerce platforms such as eBay, ratings can only be offered after transactions are completed. Since each transaction has a fee attached to it, ballot box stuffing is made expensive. Similarly, in health recommender systems, ratings can be restricted to patients who have been treated for a medical condition thus making ballot stuffing expensive.

Generally, recommender systems that rely on user ratings for recommendations tend to suffer from two major challenges; scalability and rating prediction accuracy (Yu, P et al., 2016). First, with scalability, Recommender systems incur heavy computational load as the amount of data provided increases. Consequently, the computational cost becomes extremely expensive if all user ratings are taken as input (Yu et al., 2016). Secondly, the rating prediction accuracy in recommender systems heavily relies on user ratings or preferences. Often than not, traditional methods follow the assumption that the user ratings perfectly reflect their opinions and interests. Prior studies, however, have reported that user ratings are naturally imperfect and noisy (Amatriain et al., 2009; Herlocker et al, 2004), which limits the measurable power of a recommender system. This challenge is also known as the magic barrier of recommender system (Herlocker et al., 2004).

In this paper however, we do not consider the implications of these attacks. We therefore recommend that the techniques that already exist in literature to deal with these attacks be extended to our systems (Burke et al, 2006).

2.6. Related Work

There has been considerably little research in the area of recommender systems. Consequently, most recommendation algorithms have been proposed in literature (Canny, 2002; Koren and Volinsky, 2009; Miller, Konstan & Riedl, 2004; Rendle, S. et al., 2011; Yu et al., 2014; Zhan, 2010; Berjani & Strufe, 2011; Mbandu et al., 2015; Zhuang et al., 2013).

Canny (2002) addresses the problem of collaborative filtering which can be solved via expectation-maximization, such that the update rules only require addition. Miller, Konstan & Riedl (2004) proposed PocketLens, which adressed the problem of collaborative filtering. PocketLens is a similarity-based approach which computes good ratings for items based on similarity of users, referred to as neighbours. The similarity of the user ratings is obtained by computing the similarity measure using a dot product. This way, accurate user ratings for items can be obtained using the

Zhan (2010) propose a recommendation system for computing Pearson correlation. As the researcher mentions in the paper, “the computation only requires multiplication, and is therefore relatively easier than the one we describe in this paper.”

Berjani & Strufe (2011) proposes an alternative to homophobic encryption-based approaches through data perturbation. In data perturbation, users obfuscate their data before allowing it to be used in the computation. Data obfuscation is done by adding noise to it thus ensuring that the users’ ratings are protected from manipulation.

Specifically, in healthcare, one privacy preserving recommendation system is due to Katzenbeisser & Petkovic (2008). They propose a privacy preserving recommendation system where recommendations are obtained through first encoding all relevant information such as symptoms and diseases into a standardized binary vector. A matching protocol is then used to determine the doctors with the best similar expertise through a secure matching algorithm. The most suitable result is therefore provided as a recommendation. This solves a slightly different problem than our solution, as healthcare users obtain reputation scores for optimal providers who can treat specific health condition(s), whereas the system of Katzenbeisser & Petkovic (2008) makes no such guarantees.

Mbandu A.S et al (2015) proposes a system where the private user data is encrypted using a variant of ElGamal and recommendations are generated by applying an iterative procedure based on conjugate gradient algorithm. While Mbandu prefers to work with encrypted user data, Polat & Du (2005) suggest to protect the privacy of users by using randomization techniques. In their paper, they blind the users' data with a known random distribution assuming that in aggregated data this randomization will cancel out and the data obtained will be a good estimation of the intended original data. The success of this method is highly related to the number of users participating in the computation, and this creates a trade-off between accuracy/correctness of the recommendations and the number of users. In addition to this information leakage, the randomization techniques are believed to be highly insecure (Canny, 2002).

2.7. Measuring Healthcare Quality using Recommender Systems

The definition of high-quality health care varies among individuals. For some people, quality healthcare definition revolves around whether they can go to the healthcare provider of their choice. For other people, it means access to specific types of treatment. There has been lots of attention paid to defining health care quality in recent years.

The Institute of Medicine of the National Academy of Sciences (2001) defined quality health care as "safe, effective, patient-centered, timely, efficient and equitable." Moreover, the Agency for Healthcare Research and Quality (AHRQ), the American government's leading agency charged with improving the quality, safety, efficiency and effectiveness of health care for all Americans, defines quality health care "as doing the right thing for the right patient, at the right time, in the right way to achieve the best possible results."

To improve health care quality, we need to be able to measure it (IOM, 2001). Various ways to measure healthcare quality have been proposed in previous research. One way involves measuring the ***processes of care***. Another way of measuring quality healthcare involves measuring the ***outcomes of care***. This approach focusses the patient's outcome after treatment of a health condition. A third way of measuring healthcare quality involves measuring ***the experience of patients and their family members.***

Often than not, Recommender systems rely on the ratings of previous users to make recommendations. In healthcare domain, health recommender systems rely on the experience of previously treated patients and use the patient feedback to make recommendations to other interested patients. Consequently, our proposed system relies on doctor-patient relationship, interactions in the doctor’s office, and the effectiveness of treatments offered by the health providers, where these factors are used to calculate the aggregated ratings for health providers.

According to a survey conducted by the Associated Press-NORC Center for Public Affairs Research in 2014, 59 percent of Americans indicated that the most important factor that makes a high-quality doctor is the doctor-patient relationships and physician personality. Additionally, 18 percent of the Americans said that a quality doctor is attentive, listens, or shows interest in them. Other responses focus on doctor-patient interactions and their traits, including that the physician has a caring attitude (8 percent), good bedside manner (8 percent), various other positive personality traits (7 percent), and time spent with patients (5 percent). In relation to the delivery of care or patients’ health outcomes, 11 percent valued most a doctor’s ability to diagnose accurately and fix their health problem, and 8 percent mention a knowledgeable physician.

Ashish Jha (2014) conducted a Twitter poll about what makes a good doctor. Out of 200 respondents, 18 percent said that a good doctor is empathetic. Other respondents said that a good doctor is a good listener (14 percent), compassionate/caring/kind (13 percent), Humble (9 percent), competent/effective (6 percent), intelligent (3 percent) and observant (2 percent). It is worth noting that most people believe that the doctor-patient relationship is the most important factor in differentiating quality doctors from mediocre ones, and assume that doctors meet a threshold of intelligence and are knowledgeable.

2.8. Healthcare providers evaluation metrics

Often than not, Recommender systems rely on the ratings of previous users to make recommendations. In healthcare domain, health recommender systems rely on the experience of previously treated patients and use the patient feedback to make recommendations to other interested patients. Consequently, our proposed system relies on doctor-patient relationship, interactions in the doctor’s office, and the effectiveness of treatments offered by the health providers, where these factors are used to calculate the aggregated ratings for health providers.

Our proposed Recommender system rely on various evaluation metrics that will be used to rate healthcare providers. The evaluation metrics focus on doctor-patient relationship and their traits, including that the doctor caring attitude, bedside manner, time spent with patients and other positive personality traits. Doctor evaluation metrics include such factors as Doctor Attentiveness, promptness in attendance, proper explanation of condition, care and concern, professionalism and courteousness. In addition to doctor evaluation, other evaluation metrics focus on delivery of service by the healthcare provider. Service delivery in this case include factors such as front office service. Factors in this case include Promptness/responsiveness to queries, attentiveness and waiting time to see doctor or access particular service(s). In addition to the above metrics, other important metric to measure quality healthcare providers includes the cleanliness of the office environment, including general cleanliness of the healthcare provider.

2.9. The Gap

Our work differs from previous works as reviewed in the literature in a few ways. First, most of the recommender systems are based on collaborative filtering techniques and have been widely applied in e-commerce and entertainment industries and very little research has been done in health recommender systems. We therefore propose a recommender system that takes sensitivity of healthcare information into account. In our proposed architecture, a healthcare user submits ratings anonymously to the system and thus the identity of the user is hidden. Moreover, a user who intends to obtain a recommendation from the system is able to obtain such recommendation without compromising his identity. Either way, the identity of any user interacting with the system is preserved. Lastly, whereas previous approaches use collaborative filtering technique to obtain recommendation, our proposed architecture is reputation-based and allows for the computation of real-valued rankings of healthcare service providers hence giving more fine grained recommendations.

CHAPTER THREE: METHODOLOGY

3.0. Introduction

This chapter describes the methods that were employed to achieve our proposed recommender privacy preserving healthcare scheme. This chapter starts by presenting the data collection method that was used. The chapter also contains a description of the system development methodology used, followed by an analysis and design of the system.

3.1. Data sources

The source of data will be from a set of healthcare users who will interact with the system and evaluate it. Information obtained from the healthcare users will help in evaluating the system in order to determine whether the developed healthcare system met their desired requirements.

3.2. Data collection methods

Data will be collected through the use of survey questionnaires. The questionnaires will be distributed to a set of healthcare users who will be required to fill the questionnaires. The questionnaires will be distributed through drop-and-pick method.

3.3. System Development Methodology

The proposed healthcare system uses evolutionary prototyping model for system development. This is because our system is based on designing an architecture and protocols for implementation. Our design aims to meet the objectives through understanding the requirements, both functional and non-functional, and including the requirements in our proposed prototype.

Figure 5 below illustrates the Evolutionary Development Process:

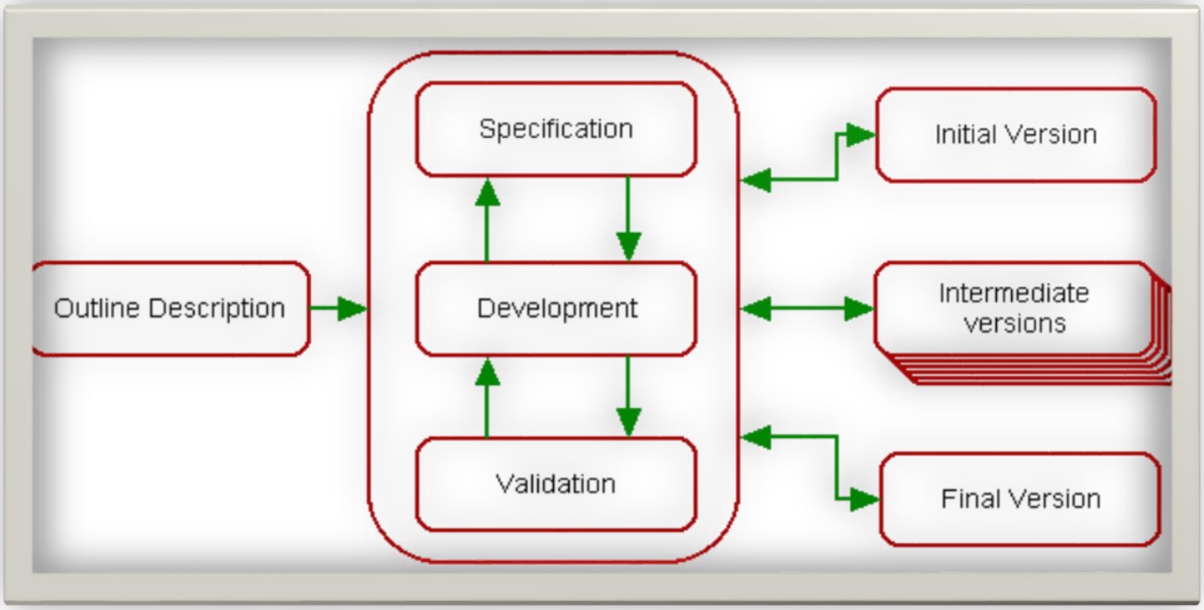


Figure 5: Evolutionary Development process

Source: Nabil A. M; Govardhan A. (2010) "A Comparison between Five Models of Software Engineering, IJCSI.

3.3.1. Specification

In this phase, we defined the functional and non-functional requirements for our proposed architecture. We also defined the assumptions and limitations of our research work.

3.3.2. Development

The requirement specifications will be studied, and a healthcare recommender system architecture developed to be used for implementation of algorithms for this system.

3.3.3. Validation

This phase in our research work will include evaluation of the implementation system and an analysis of the algorithms for performance. Such analysis will entail security analysis for the protocols (algorithms) designed for implementation of our proposed privacy-preserving health recommender system. 3.4. Our Proposed Architecture

3.4. System analysis

The purpose of the analysis phase is to produce a set of roles whose tasks describe what the system has to do to meet its overall requirements. A role describes an entity that performs some function within the system. We gathered the requirements by first identifying the desired system inputs and outputs, and then studying the user's environment and identifying the goals of the system. This enabled us in one hand to identify the use cases for the new system and on the other hand to model them. The use cases modeled were then specified as user requirements. We next built the class diagram to realize every use case in the diagram. Once all use cases are realized in the class diagram, we transformed the identified goals into a set of roles and built role model diagram.

3.4.1 Inputs and outputs

3.4.1.1 Inputs

The system will require a healthcare user to submit ratings for services sought from a healthcare provider. The user will be presented with a five-point rating scale on which to rate a healthcare provider for the treated health condition. A healthcare user can also issue a Reputation Request to the system so as to receive recommendations. Therefore, there are two inputs made to the system; user feedback, and Reputation Request.

3.4.1.2 Outputs

The system will compute the reputation score from the available user ratings by finding a weighted average of the ratings. The system will thus return the names recommended healthcare providers in the form of a list starting for the best rated healthcare providers of the specific health condition, and can therefore enable the healthcare user make an informed decision on the preferred reputable healthcare provider.

3.4.2 Identifying goal

Goal identification is the first step in the analysis phase, which takes an initial system specification and transforms it into a structured set of system goals. During analysis, this process involved capturing the system goals and then structuring the goals into a hierarchy based on their importance and level of detail.

a) Capturing Goals

This process involved extracting scenarios from the initial specification and describing the goal of that scenario. The following are the scenarios from our initial specification:

- The system is responsible for making recommendations for the optimal healthcare service providers to healthcare users.
- A previously treated healthcare user will be able to provide feedback by submitting ratings for the health conditions treated by the healthcare providers.
- A healthcare user who wishes to obtain recommendations will be able to obtain such from the system.
- The system will make recommendations based on the computed reputation score and make the recommendations available to the user.

From the above scenarios, the derived system goals are

1. Receive user feedback (ratings).
2. Store user feedback in database.
3. Perform rating aggregations.
4. Compute the reputation score of healthcare providers.
5. Make recommendations for optimal healthcare provider

b) Structuring the goals

After capturing of goals, the goals structured into a hierarchy depending on the importance and level of detail. Figure 6 shows the goal structure diagram.

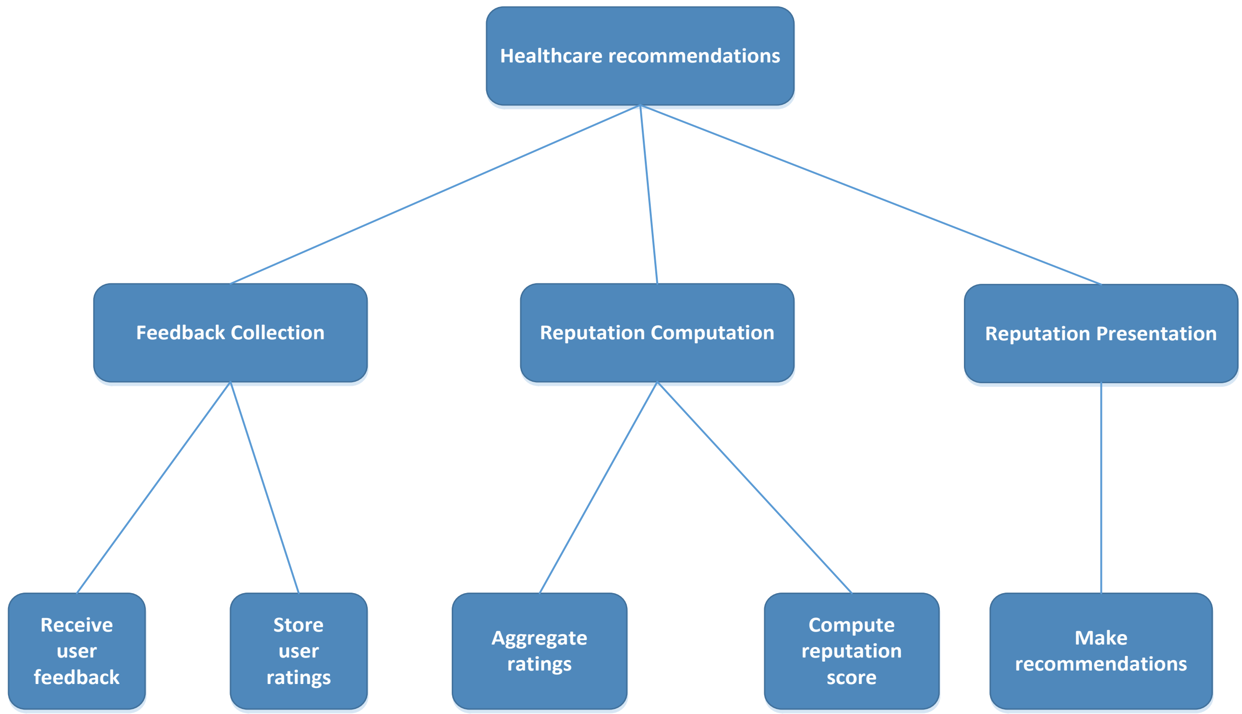


Figure 6: Goal structure diagram

3.4.3 UML use case diagrams

Use case diagrams describe the functionality of a system and the users of the system. The use case diagrams consist of actors and use cases. Use cases are the services provided by the system to the actors (users). Use case diagrams for each entity present in the system is presented in Figure 7. These include use case diagrams for the Healthcare user and the Recommender class.

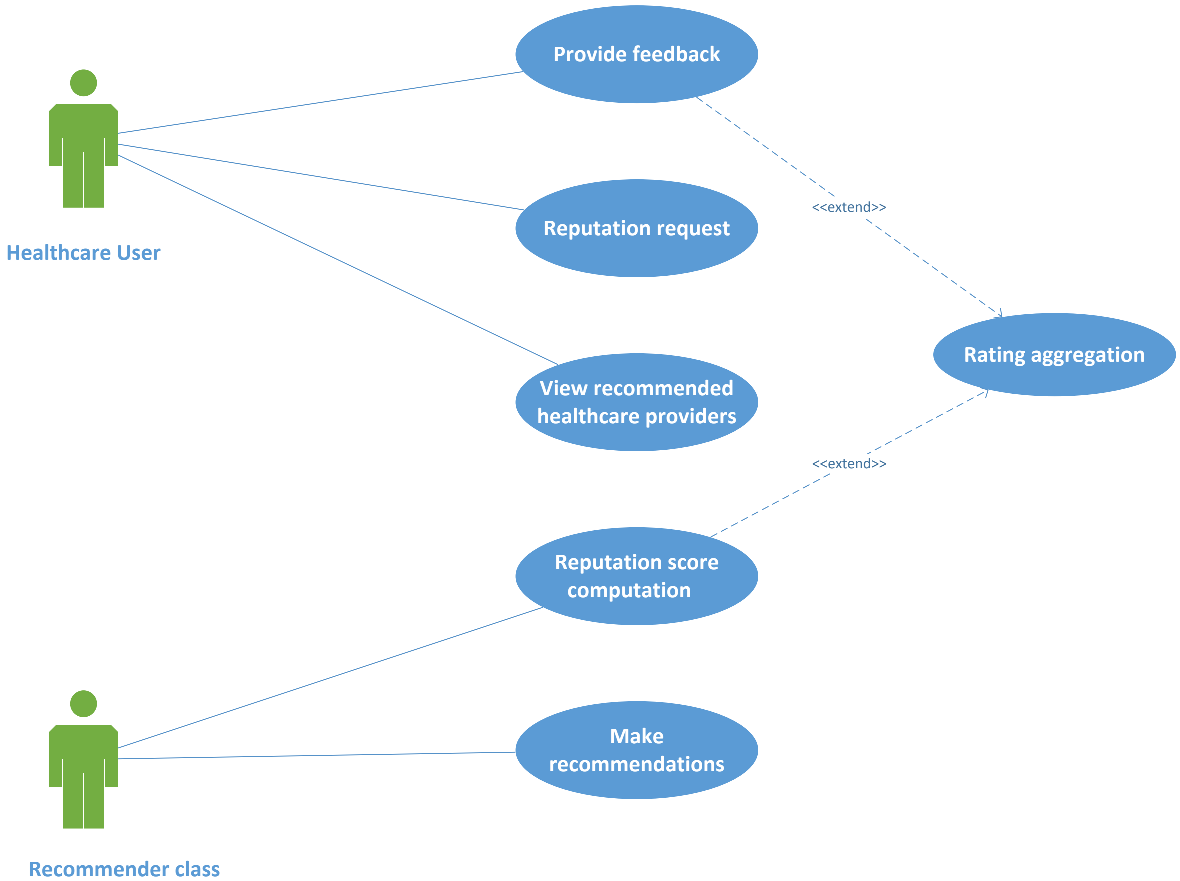


Figure 7: Use case diagram

3.4.4 Refining roles

We built a role model diagram to transform the structured goals and use cases into roles and their associated tasks. Figure 8 shows the role model diagram.

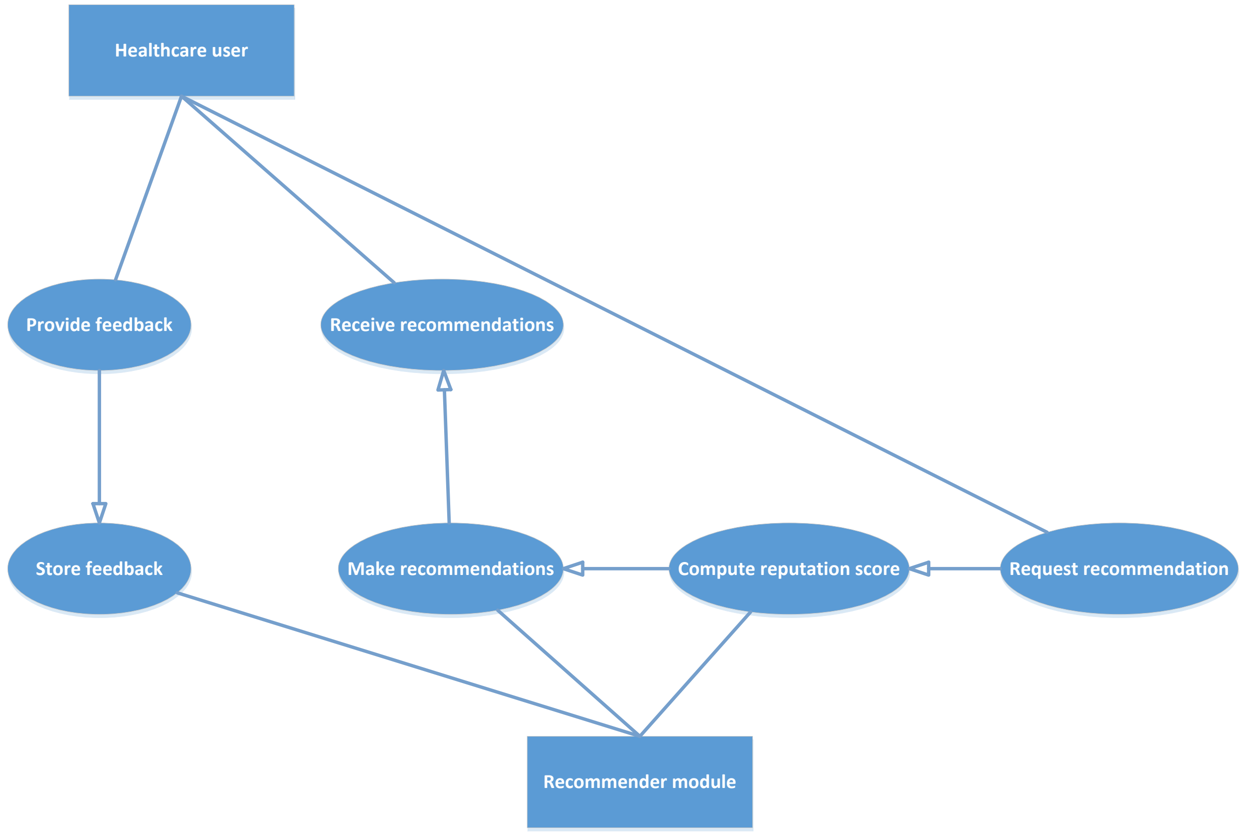


Figure 8: Role model diagram

3.4.5 Class diagram

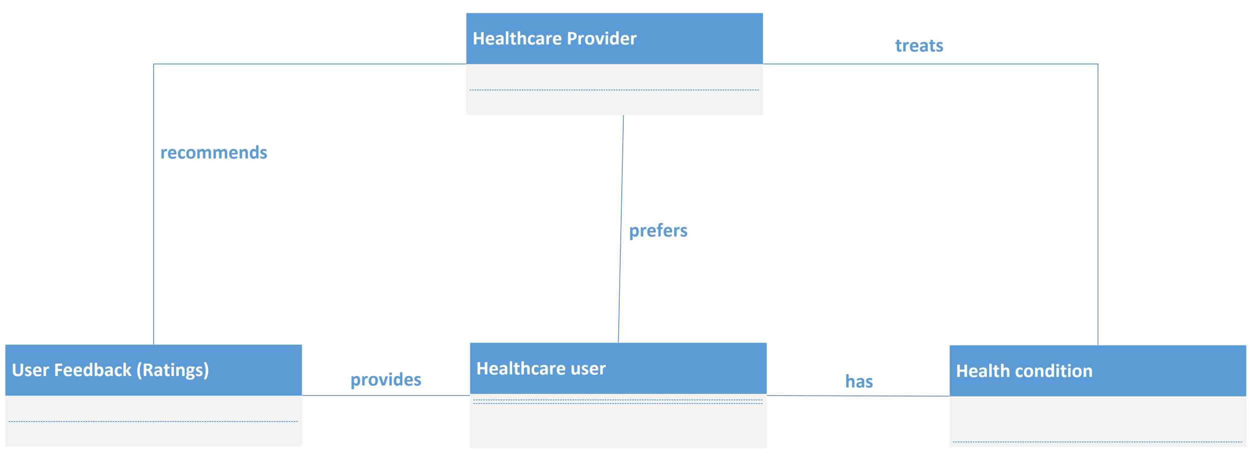


Figure 9: Class diagram

3.4.6. Requirements Analysis

After identifying the use cases for the healthcare system and modelling them, we gathered the functional and non-functional requirements.

3.4.6.1.Functional Requirements

The proposed recommender reputation-based healthcare system must conform to the following requirements;

i. A healthcare user interested in a particular health condition should be able to obtain a healthcare provider recommendation for the condition based on the aggregated ratings from previous healthcare users who accessed that particular service being sought by the requesting user. Moreover, the recommendation should include alternative best-ranked healthcare service providers instead of providing only a single recommended one. A healthcare user requesting a recommendation will therefore not only be presented with the name of the best service provider, but also a list of alternative best providers from which to obtain service.

ii. A healthcare user willing to provide feedback for reputation score computation should be able to do so and the system should provide a platform for the user.

iii. The reputation of the healthcare providers in the system must be preserved, or at least the effectiveness of a small number of malicious users in altering heath providers' scores must be mitigated.

iv. The recommender system must preserve the privacy of the healthcare user requesting for recommendation.

Table 1 shows the functional requirements described in shape of use cases.

Primary actor	**Use cases**
Healthcare user	Provides feedback via ratings for healthcare providers and health conditions
	Gets healthcare recommendation based on aggregated user ratings

Recommender class	Computes the reputation score based on aggregated user ratings
	Makes recommendation based on top-N recommended healthcare providers

Table 1: Functional requirements

3.4.6.2.Non-functional Requirements

A new user should be able to use the recommender engine without putting too much efforts on learning how to use it, and, in case of doubt, there must be some help to solve their doubts.

3.5. System Design

3.5.1 Overall System Architecture

Architecture is created to describe the structure of the system to be built and how that structure supports the business and service-level requirements.

The system will be in the form of a client-server architecture. The client is the front end web-based API that a healthcare user is presented to interact with the system. On the other hand, the server-side entails the recommender module, feedback module and the database. A healthacare user can provide feedback or query the system for healthcare recommendation. Feedback provided by a user in the form of ratings through the feedback module will be stored in a MySQL database. A user who needs a reputation recommendation will give a reputation request to the recommender system through an anonymizer. This way, the identity of the user querying the system for recommendation will be concealed and the system wil learn no information from the user. Anonymizer systems are all around us, and have been used for anonymization. An example of an anonymizer system that has been widely used is the Tor anonymizer network. The anonymizer will therefore query the recommender system on behalf of the user and therefore the system is unable to make any inferences between the requesting user and the specific health condition that a user requires recommendation.

At present, when healthcare users need to access healthcare services, they rely on friends and family on recommendations. These friends or family may not have been treated for the health condition the healthcare user is interested in. Moreover, recommendations by a few individuals may not guarantee an accurate recommendation. The healthcare system will provide a list of recommended healthcare service providers based on the aggregated user ratings. A new user who wishes to interact with the system can be able to do so either by submitting a rating or by requesting a recommendation.

The system will make healthcare recommendation available to users by calculating the weighted average ratings based on the combined reputation score. Only the user who has accessed a service from a healthcare provider can be allowed to provide a rating for the health condition treated by that healthcare provider.

We now present the architecture of our proposed recommender privacy preserving reputation based healthcare system.

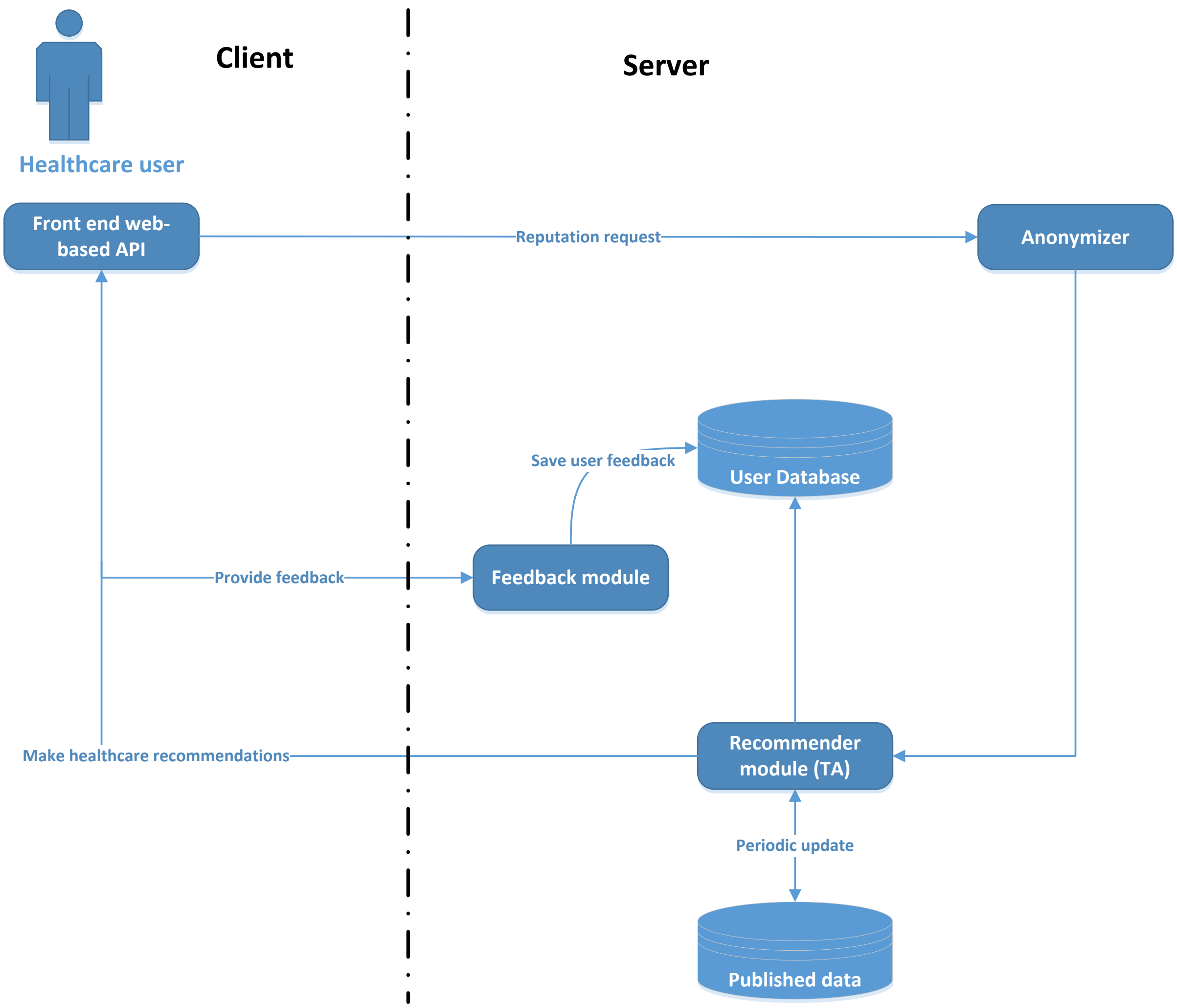


Figure 10: system architecture.

3.5.2 Flow Design

The overall system flow is shown in figure 11.

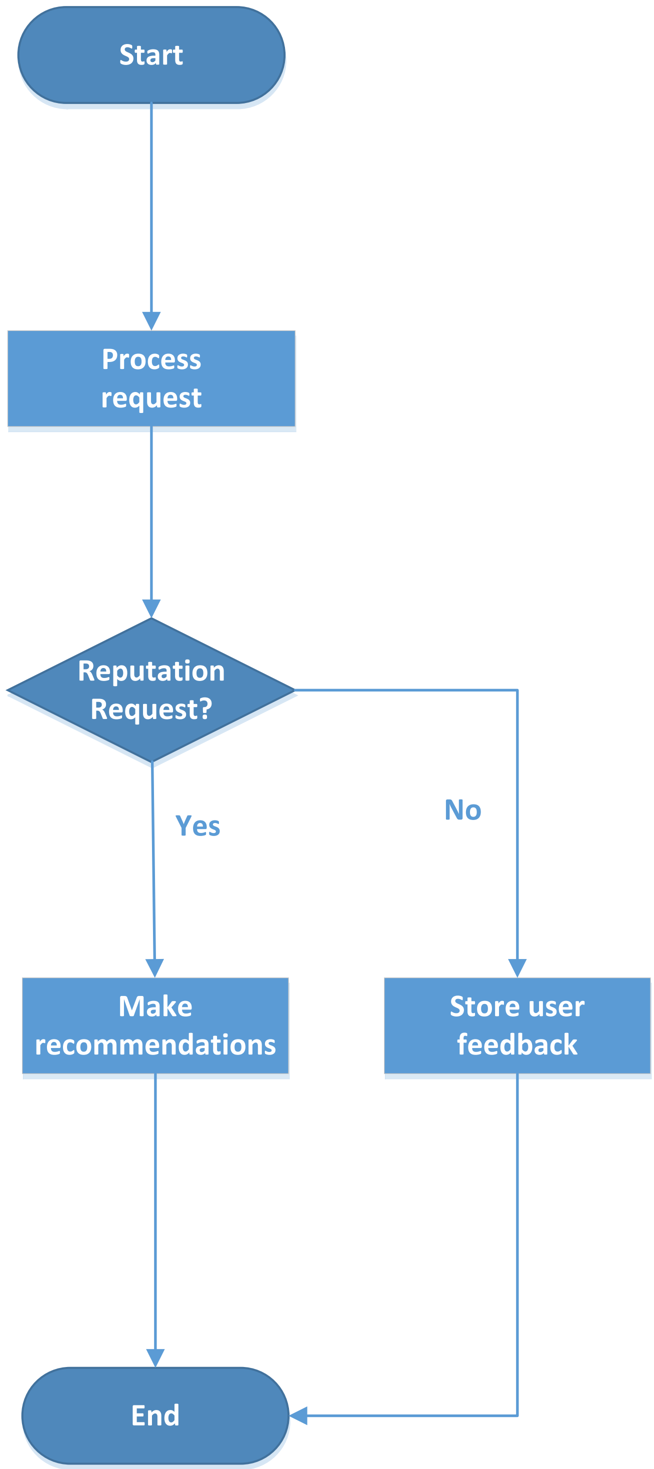


Figure 11: System flow chart

3.5.2.1 Healthcare user providing feedback

Figure 12 shows the process flow of the activities of a healthcare user who wishes to provide user feedback.

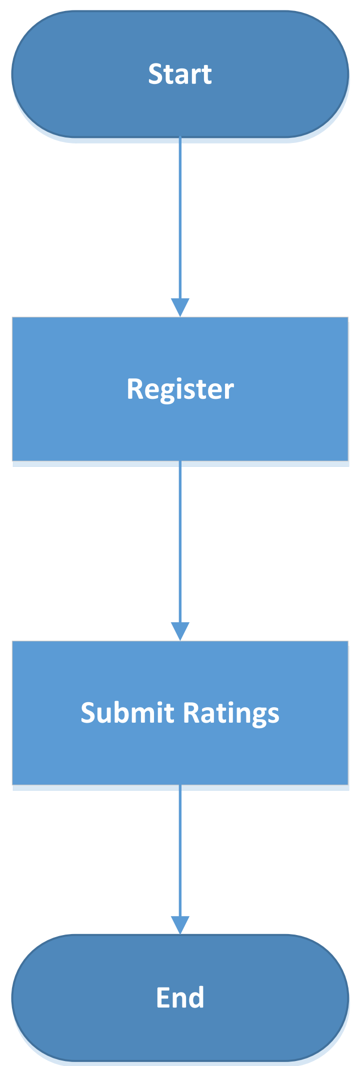


Figure 12: Healthcare user submitting ratings process flow

3.5.2.2 Healthcare user requesting recommendations

Figure 13 shows the process flow of the activities a healthcare user performs when querying the system for reputation score.

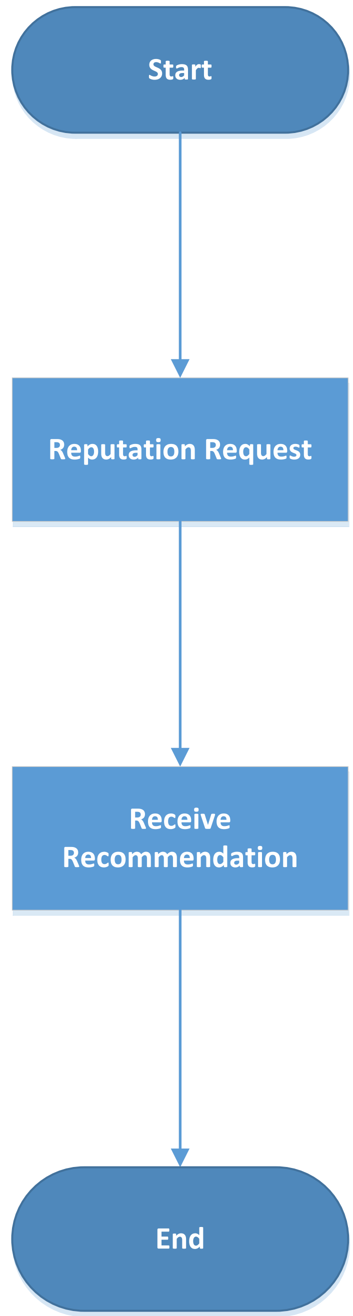


Figure 13: Healthcare user requesting reputation recommendation process flow

3.5.3. Sequence Diagram

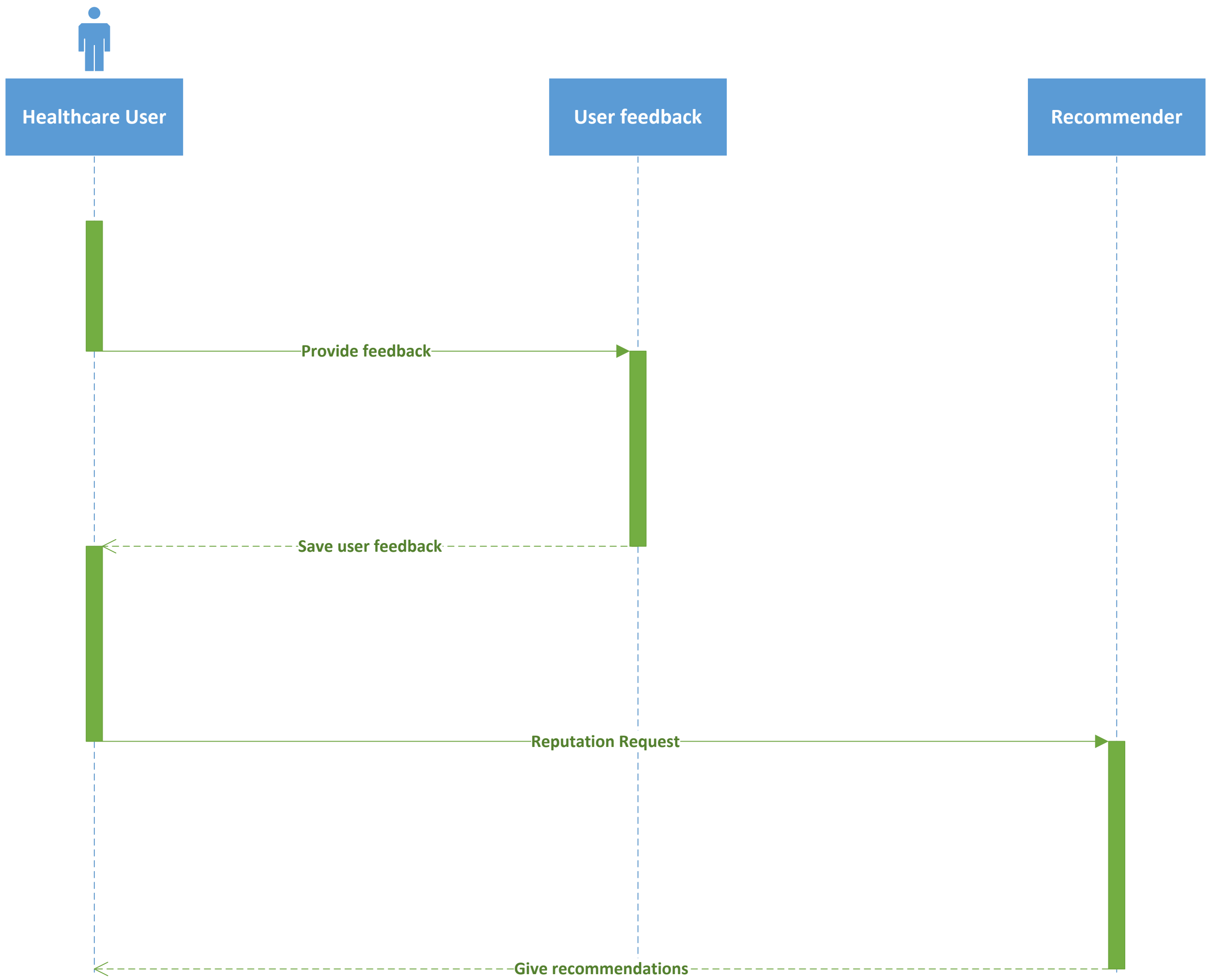


Figure 14: Sequence diagram

3.5.4. Database Design

The database stores the information about the healthcare provider ratings and acts as a repository for user feedback as provided by the users. The system database was built using MySQL database. Database creation and management was achieved using PHP MyAdmin which is a component of XAMPP server. The database is called **ratings** and it contains the **rating_details** table. The ratings_details table would contain health condition and healthcare provider information and also the rating metrics for the specific health providers and the corresponding health conditions. Doctor's care and concern among others. Other important information contained in the rating_details table is the email information for the healthcare users providing feedback to the system. Email information for users helps prevent the users from stuffing the recommender system with repetitive ratings for the same health condition to the system. This way, a healthcare user can only rate a healthcare provider treating a particular health condition only once.

In a nutshell, the database consists of the ratings table.

3.6.4.1 Rating Details Table

Name	**DataType**	**Length**
Rating_id	int	15
Email	varchar	50
Hospital	Int	2
Condition	Int	2
Doctor_promptness_attandance	Int	2
Doctor_listens_attentive	Int	2
Doctor_proper_explanations	Int	2
Doctor_care_concer	Int	2
Doctor_professional_courteous	Int	2
Reception_promptness_attendance	Int	2

Reception_polite_friendly	Int	2
Reception_wait_service	Int	2
Cleanliness_facilities	Int	2
Average_rating	Int	2
Date_provided	Int	2

Table 2: Ratings details table

3.6. System Implementation

3.6.1. System Development

The system will be implemented in PHP (Hypertext Preprocessor) and the following prerequisites will be needed in order to run the application correctly.

- Apache Tomcat webserver – this is a webserver to aid healthcare users to access the system through a web interface.
- MySQL – This is the data store where all the information will be stored.

The development was done in modules and the below modules were identified

3.6.1.1 The patient feedback module

This is the feedback collection module that is used by the healthcare user to submit ratings to the system for reputation computation.

3.6.1.2 The Recommender module

This is the module that takes the healthcare user feedback, calculates the aggregate ratings for healthcare providers and the corresponding health conditions they treat, and then makes recommendations to interested users for the top N rated healthcare providers for a specific health condition.

3.6.1.3 The Reputation presentation module

This module presents a user with a list of optimal healthcare providers that can treat specific health conditions.

3.6.2. System Configuration

After the system have been developed the following needs to be performed

- Installation of Apache tomcat web server
- Installation of a MySQL database.

3.6.3. Ratings aggregation process

Our method can be described as weighted average where the weights are generated based on the vitality of the evaluation metric in providing quality healthcare. For example, doctor-patient relationship is viewed as a mire important factor in determining quality of healthcare as compared with the other factors. As such, in our case, our proposed recommender system provides different weights for ratings, where the more important the evaluation metric is, the higher the weight the metric will get. In other words, using this weighting method we can assign higher weights to highly rated evaluation metrics, which we believe will generate more accurate recommendations.

3.6.4. System experimentation/interaction

The implementation of the healthcare recommender system will run as follows:

1) A healthcare user with the need for a reputation score of a certain heath condition k, which is treated by healthcare provider p will make a Recommendation Request RRQ to the Healthcare system.
2) The Recommender module will perform the reputation computation by aggregating ratings of the healthcare users who had previously been treated by the health providers.

3) The Reputation presentation module then presents the Healthcare user who issued a Reputation Request with a list of optimal healthcare providers that can treat specific health conditions.
4) Our Healthcare Recommender system relies on previous Healthcare users who were treated specific health conditions by particular Healthcare providers. As such, a healthcare user wishing to submit ratings to the system can be able to do so through the feedback module. The feedback module provides the user with a list of health conditions and the corresponding healthcare providers from which they will provide feedback to the system. The user feedback is converted and stored in a database in the form of ratings.

CHAPTER FOUR: RESULTS AND DISCUSSION

4.0 Introduction

This chapter presents the results of the data analysis. The chapter also includes the outputs of the health recommender system, the system interface and the test results.

4.1. System output

The system computes the reputation score from the available user ratings by finding a weighted average of the ratings. The system them provides healthcare recommendations to interested users based on specific health conditions as treated by healthcare providers. This is done by the reputation presentation module. The system thus returns the names of recommended healthcare providers as output, in the form of a list starting for the best rated healthcare providers of the specific health condition, and can therefore enable the healthcare user make an informed decision on the preferred reputable healthcare provider.

4.2. System Interface

4.2.1. Submitting patient ratings

This is done by the feedback module which presents patients with a set of queries for rating the healthcare providers. A healthcare user will be presented with a five-point rating scale on which to rate specific attributes of a healthcare provider treating particular health conditions.

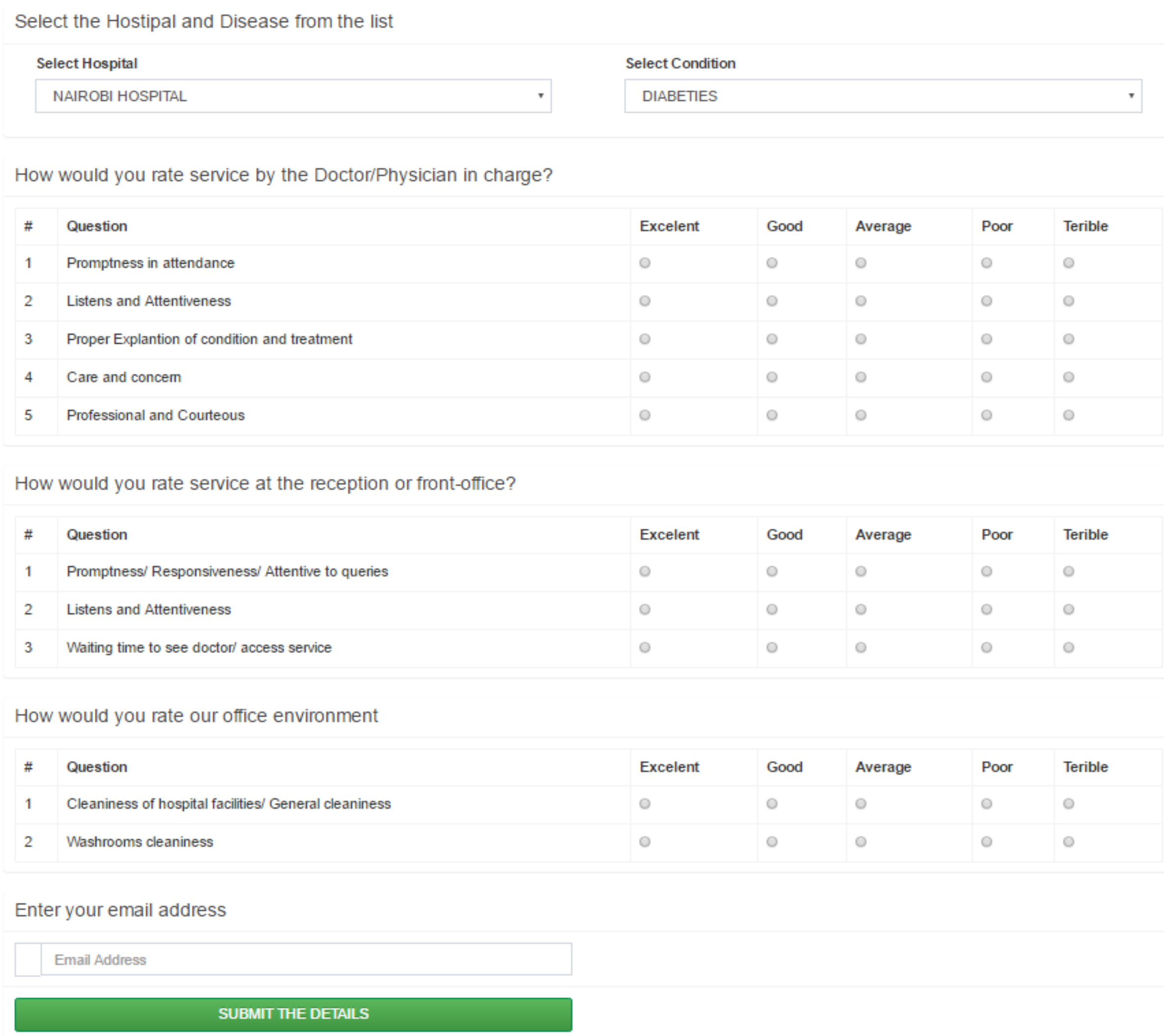


Figure 15: The feedback module

4.2.2. Providing healthcare recommendations

The system provides healthcare recommendations to interested users based on specific health conditions as treated by healthcare providers. This is done by the reputation presentation module. The system thus returns the names of recommended healthcare providers in the form of a list starting for the best rated healthcare providers of the specific health condition, and can therefore enable the healthcare user make an informed decision on the preferred reputable healthcare provider.

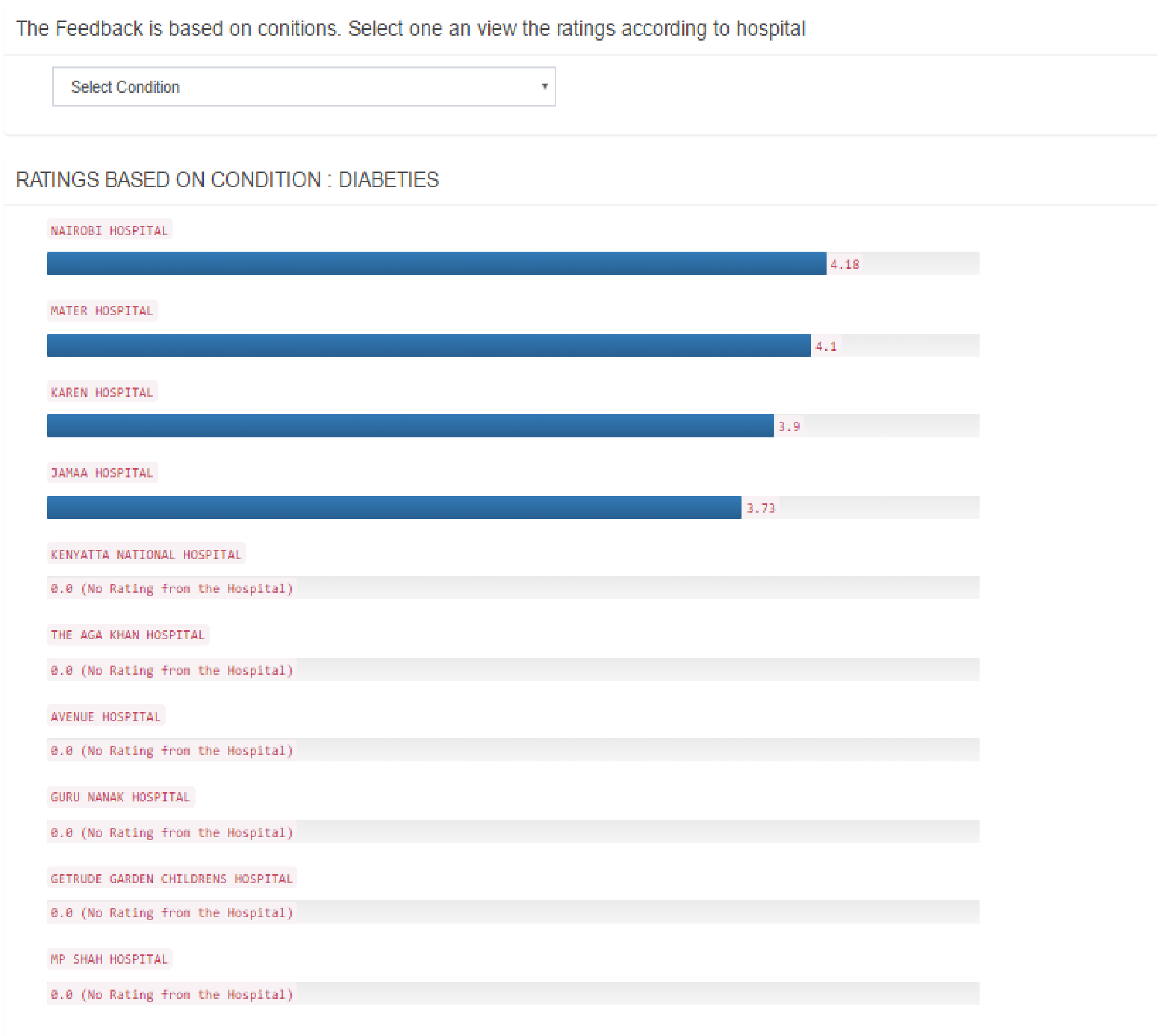


Figure 16: The reputation-presentation module

4.3. System evaluation

System evaluation was done by a set of healthcare users who interacted with the system and gave their input in a questionnaire. A survey was done to healthcare users who evaluated the system and the survey involved the respondents answering eight questions, of which seven were multi-choice question and the eighth was an open question requiring a general response. The survey contained seven statements in which the healthcare users stated how much they agree with the recommendation made by the system in a scale of 0 to 4. '0'represented "Not Applicable",'1' represented “Not useful” whereas ‘4’ represented “very useful”. In addition to that, the system has an online review module where users can provide feedback on how useful the recommendations received were, based on their experience with previously recommended healthcare providers. Lastly, this chapter provides an overview of the system testing strategies carried out on the developed healthcare recommender system.

We reviewed survey responses from 50 participants, and the following are frequencies of the responses from the participants. Below are the seven statements and the frequency of each score for the 50 respondents

		User review Frequency for each review Question				
		Not Applicable	**Not Useful**	**A bit useful**	**Useful**	**Very Useful**
Review Questions	How do you Rate The overall recommendations made by the system?	0	4	6	32	8
	How easy to use is the system?	2	2	8	18	20
	Would you recommend the system to other healthcare users?	2	5	6	14	23
	Would you use the system again/ another time to provide feedback or obtain recommendations?	1	4	8	16	21

	How relevant were the recommendation made for healthcare providers by the system?	0	2	11	13	24
	How relevant were the recommendation made for health conditions by the system?	0	2	10	15	23
	How useful was the feedback you submitted for providing recommendations?	1	4	9	17	19

Table 3: System healthcare users' review summary

The first four questions were intended to capture an overall perception of the system by the healthcare users. This is presented graphically in the figure below. And from the data collected the number of healthcare users who found the system as "Useful" And "Very useful" out ways those that found the system as "Not useful" and this shows that the users have a positive perception about the system and if the system is fully implemented they are likely to use it to provide feedback and obtain recommendations for optimal healthcare providers.

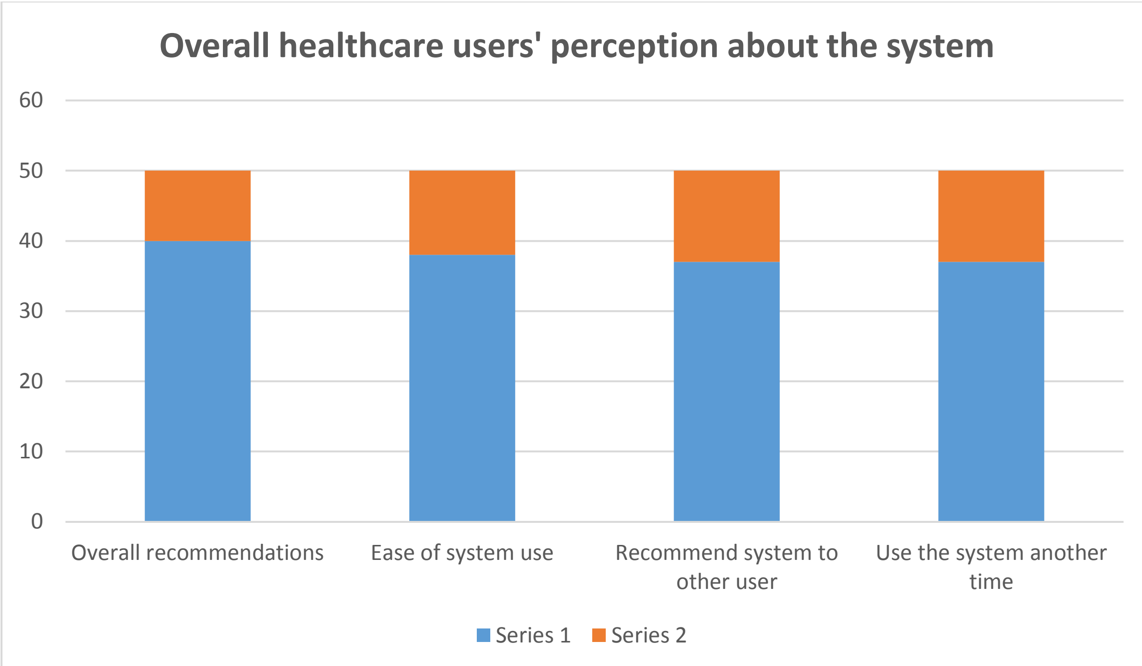


Figure 17: Overall healthcare users' perception about the system

From figure 18 above, Series 1 represents the healthcare users who rated the system as either "Useful" or "Very Useful" and series2 represents the users who rated the system otherwise.

The next three questions were intended to measure how useful the recommendations were to the healthcare users and from the graphical representation its shows most users found the healthcare recommendation provided by the system objective and relevant to their expectation which is an impression that the system was useful to them and chances are that they will use the fully implemented system.

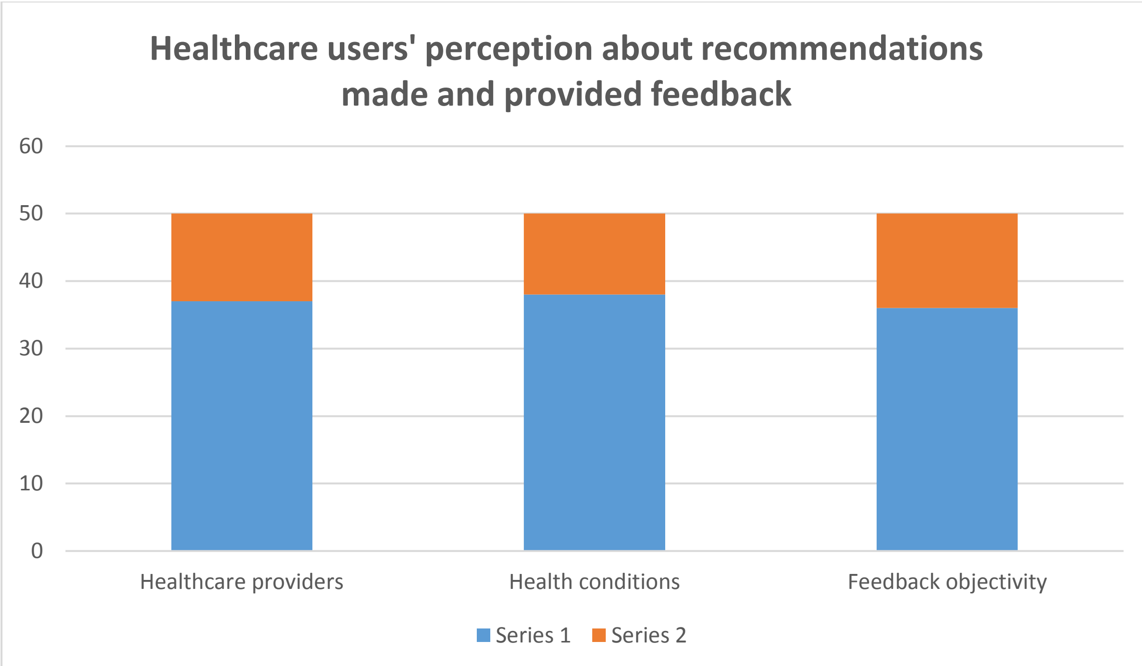


Figure 18: Healthcare users' perception about recommendations

From figure 19 above, Series 1 represents the healthcare users who rated the system as either "Useful" or "Very Useful" and series 2 represents the users who rated the system otherwise.

In addition the system has an online feedback option for users who opt to provide online reviews for the recommendations provided by the system. Below is a graphical representation of the feedback provided by the 32 learners who opted to provide a review on how useful the recommendations received from the system were, based on their experience with previously recommended healthcare providers.

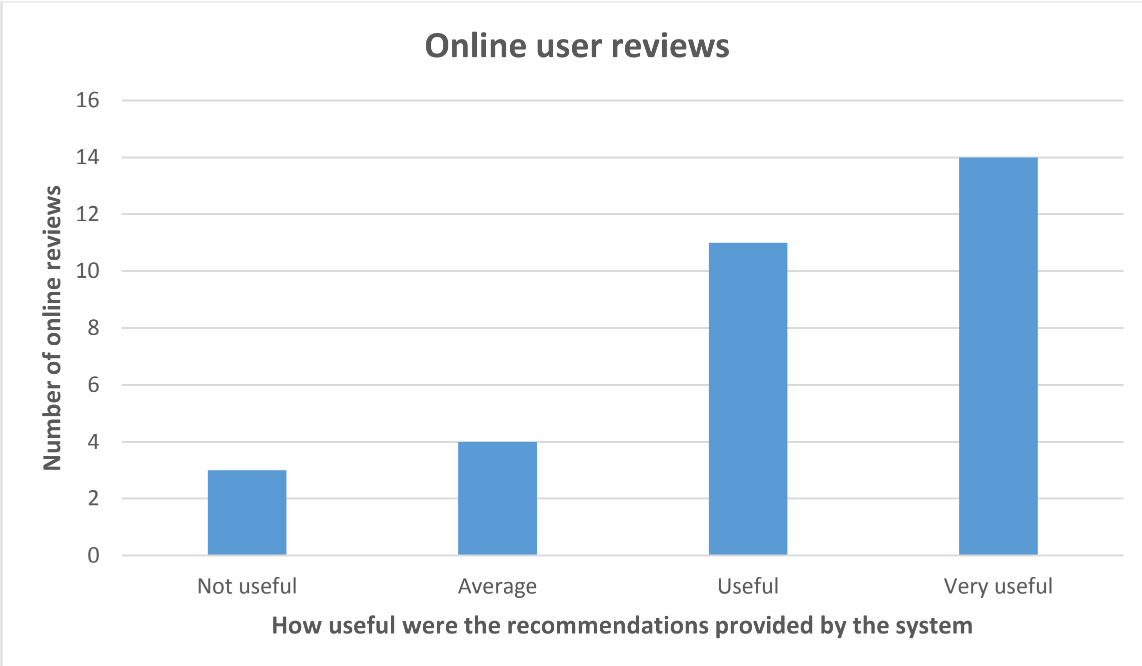


Figure 19: online Use Reviews

The last question was an open ended question which required the participants to state their general perception about the system and whether they think that the system will help users in obtaining quality care from healthcare. Out of the 15 respondents who provided reviews for the other questions only 7 of them opted to respond to this question and the below are their responses.

1. The system will help promote assess to quality healthcare to healthcare users.
2. They system will be a good tool to use when searching for optimal healthcare providers on specific health conditions.
3. The system will help popularize healthcare providers who offer quality healthcare.
4. The number of healthcare providers and health conditions covered by the system needs to be increased so that the system can claim to represent the global space.
5. I like the system, when will the system be implemented in Kenyan health sector?
6. The system is good, but can it be modified to work offline when there is no internet?
7. This is a good system and needs to be fully developed into a production system and adapted by the health sector.

Five out of the seven respondent provided a positive review of the system and this is may be an indication of their intent to use the system once it is fully deployed.

4.3.1. Summary

The study reveals that reaction by the users towards the system is good, with about 75% of the reviewed participants responding positively. This means that the expected positive impact of the system is high and measures needs to be put in place so as to maintain this positive impact and minimize the small negative impact that has been raised by some of the reviewers.

4.3.2.. System Testing

System testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets specified requirements. System testing, in this case, involved performing a variety of tests on the system to evaluate its behavior as defined by the scope of the project. The main reason for conducting system testing was to verify the system against specified requirements. The system was checked to determine whether it was behaving as per expectations.

4.3.2.1. System Testing Results

A test case is usually a single step, or occasionally a sequence of steps, to test the correct behaviour/functionality and features of an application. An expected result or expected outcome is usually given.

Table 3: System Testing Results

The table below shows a list of test cases that were used to conduct system testing.

Test Case No	Task	Expected Results	Actual Results	Status
TC1	Installation of the system	The System should install successfully	The system is successfully installed	Pass
TC2	Healthcare user provides feedback to system	The system directs the user to their respective interface	The system user successfully provides feedback to the system in the form of ratings	pass
TC3	Storage of user feedback	Upon submitting ratings, the system stores the user feedback in a database	The system successfully stores user feedback.	Pass
TC4	Healthcare user issues a Recommendation Request	The system presents a list of top N recommended healthcare providers to user based on user's interest.	The system successfully provides a list of optimal healthcare providers diagnosing particular health conditions	Pass
TC5	Perform rating aggregation	System performs a rating aggregation of individual user ratings	System successfully performs a rating aggregation	Pass
TC6	Compute the reputation computation score	The system should compute the reputation score of the health providers whose feedback had been earlier provided.	The system computes the reputation score of healthcare providers.	Pass
TC7	Ensure healthcare user privacy	The system protects the privacy of the users interacting in the system.	The system preserves privacy and user not required to provide personal information.	Pass
TC 8	System protecting the reputation of the healthcare providers.	The system protects the reputation of the healthcare providers by preventing malicious users from health providers' scores.	The system protects the reputation of the healthcare provider. . User submitting ratings provides email information to prevent repetitive submission of ratings.	Pass
TC9	Healthcare user selecting health condition and healthcare provider.	System maintains a list of healthcare providers and health conditions from which to obtain recommendation.	System maintains a list of healthcare providers and health conditions from which to obtain recommendation.	pass

Table 4: System Testing Results

The table 5.1 above shows a list of use cases used to conduct system testing, tasks carried out, expected and actual results. From the system testing results, all tasks carried out during the test passed the test.

4.3.3. Validation testing

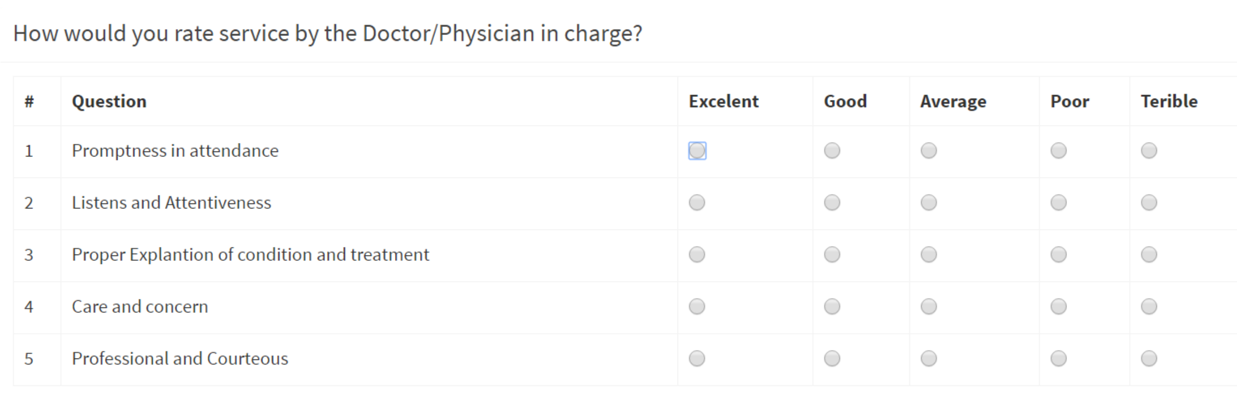
Software validation is the process of testing software to check whether it satisfies the customer needs or not. This testing is done during and/or at the end of the process of software development. The following tasks were validated during validation testing: partial feedback validation, email validation and repetitive ratings validation.

4.3.3.1 Validation testing results

The following screen shots have been used to show validation testing results;

a) Partial Feedback validation

A healthcare user providing feedback to the system can only submit all the ratings for the information to be accepted and stored in the database.



How would you rate service by the Doctor/Physician in charge?

#	Question	Excelent	Good	Average	Poor	Terible
1	Promptness in attendance					
2	Listens and Attentiveness					
3	Proper Explantion of condition and treatment					
4	Care and concern					
5	Professional and Courteous					

Figure 20: Partial feedback validation

b) Repetitive ratings validation

The system prevents user from submitting repetitive ratings to the healthcare system. This way, a user can only rate a healthcare provider treating a certain condition only once. When the user attempts to submit repetitive ratings, the system issues an alert that “user is not allowed to give the same hospital and condition twice”.

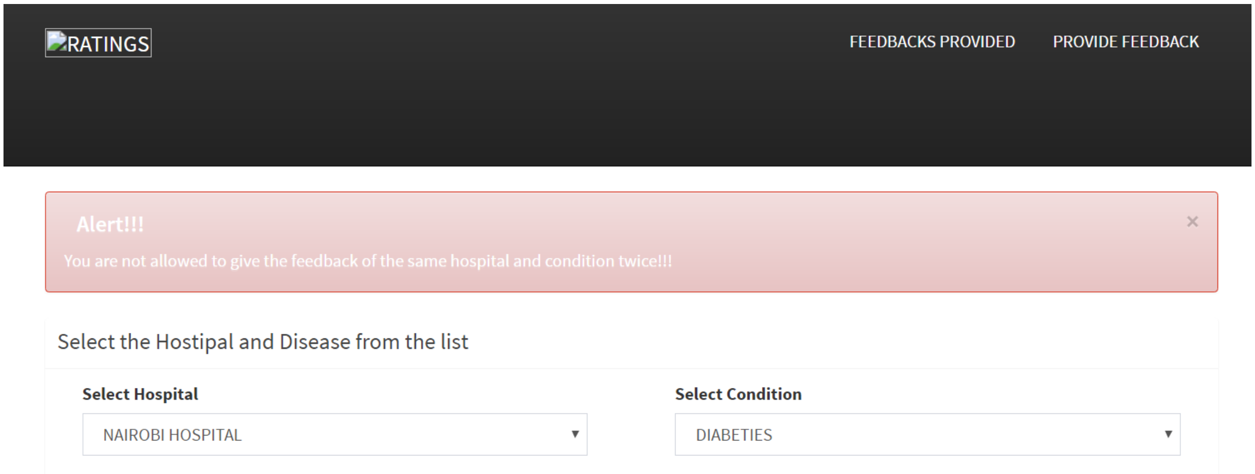


Figure 21: Repetitive ratings validation

CHAPTER FIVE: CONCLUSION AND RECOMMENDATION

5.1. Conclusion

In this study, we studied reputation systems in detail and reviewed how recommender systems make use of reputation models to enhance the quality of recommendations. In general, we noticed that available reputation models lack one or more important factors. Most of the recommender systems are based on collaborative filtering techniques and have been widely applied in e-commerce and entertainment industries and very little research has been done in health recommender systems.

Reputation-based systems provide users with advice on the quality of items on the web, based on the aggregated value of user-based ratings. Recommender systems have been widely used in many facets of life. However the same is largely lacking in healthcare domain. In this paper, we proposed a method to combine recommender and reputation-based systems to enhance the accuracy and reliability of recommendations. Also to that, we proposed a novel architecture for a recommender privacy-aware reputation based healthcare system. The architecture ensures that the privacy of a healthcare user interacting with the system is preserved. Furthermore, the proposed reputation-based system provides recommendations of the optimal health providers to healthcare user. This paper suggested a client-server architecture where a user can interact with health recommender system through an anonymizer. The anonymizer queries the system for recommendations on behalf of the user and the system is unable to make any inferences between the requesting user and the specific health condition that the user is interested in.

In our study, we planned to provide an alternative method for the proposed recommender system by developing a recommender reputation-based system for use in healthcare. This has helped us capitalize on user feedback and utilizing the feedback to obtain more accurate recommendations. Also, combine combining recommender systems with reputation systems helped us achieve synergy and avoid the weaknesses that each type individually has.

5.2. Limitations of the study

The developed system is limited to just the healthcare domain. The system is limited to be used by two sets of users only; a healthcare user interested in obtaining recommendations for specific health conditions and a user providing feedback to the system in the form of ratings. Moreover, user feedback is provided in the form of ratings and ratings are numeric in nature. This enables us to assume that an “average” rating makes sense, and is consistent across the recommendation system. In addition to that, while there are some known challenges with recommender systems such as ballot stuffing and shilling attacks (where a healthcare provider attempts to sabotage a competitor to make themselves better), we recommended that the techniques that already exist to combat these attacks be extended for our system. Finally, we made an assumption that the system maintains a list of healthcare providers and health conditions for which recommendations can be provided. This way, healthcare users will be able to submit ratings from the list of healthcare service providers provided based on specific health conditions.

5.3. Recommendation

Our reputation-based system depend on numeric data available in the healthcare domain. The numeric data is in the form of ratings by healthcare users. Other domains like e-commerce depend on ratings in, the number of likes, shares, and followers in social media, citation counts in digital libraries, or other data. On the other hand, most websites allow customers to add textual reviews to provide detailed opinion about the product (Tian, Xu, Li, Abdel-Hafez, & Josang, 2014a, 2014b). These reviews are available for customers to read, and users’ now depend increasingly on reviews rather than ratings. In our future work, we intend to use sentiment analysis methods to extract users’ opinions and use this data in our proposed system. Moreover, In the future we plan to publish a detailed survey to cover the weakness and strength of the available health recommender systems and to give more attention to the online recommender systems. We also plan to study the implications of the proposed models on industry. Also to that, we wish to implement this system on mHealth systems based on cloud computing.

REFERENCES

Abdel-Hafez, A., Xu, Y., & Josang, A. (2015). *An accurate rating aggregation method for generating item reputation*. In the 2015 IEEE International Conference on Data Science and Advanced Analytics (IEEE DSAA'2015), IEEE.

Abdullah, N. (2012). *Integrating collaborative filtering and matching-based search for product recommendation.* (Doctor of Philosophy), Queensland University of Technology, QUT ePrints.

Amatriain, X., Pujol, J.M. and Oliver, N. (2009) I like it… I like it not: evaluating user ratings noise in recommender systems. *Houben, G.-J., McCalla, G., Pianesi, F., Zancanaro, M. (eds.) UMAP 2009. LNCS, Springer, Heidelberg.* vol. 5535, pp. 247–258.

Apnorc.org. (2016). *Finding Quality Doctors: How Americans Evaluate Provider Quality in the United States: Research Highlights | APNORC.org* . [online] Available at: http://www.apnorc.org/projects/Pages/HTML%20Reports/finding-quality-doctors.aspx [Accessed 29 Jul. 2016].

Apnorc.org. (2016). *Finding Quality Doctors: How Americans Evaluate Provider Quality in the United States: Research Highlights | APNORC.org* . [online] Available at: http://www.apnorc.org/projects/Pages/HTML%20Reports/finding-quality-doctors.aspx [Accessed 5 Aug. 2016].

BANKOVIC, Z. et al. (2011). Detecting bad-mouthing attacks on reputation systems using self-organizing maps. *Computational Intelligence in Security for Information Systems.* LNCS Series, vol. 6694. Pp. 9–16.

BERJANI, B. AND STRUFE, T. (2011). A recommendation system for spots in location-based online social networks. *EuroSys Workshop on Social Network Systems*.

Blogs.sph.harvard.edu. (2014). *What makes a good doctor, and can we measure it? | An Ounce of Evidence | Health Policy*. [online] Available at: https://blogs.sph.harvard.edu/ashish-jha/2014/03/20/what-makes-a-good-doctor-and-can-we-measure-it/ [Accessed 11 Aug. 2016].

Burke, R. (2007). Hybrid web recommender systems. In P. Brusilovsky, A. Kobsa, & W. Nejdl (Eds.), *The Adaptive Web,* LNCS 4321, 377–408. Berlin Heidelberg, Germany: Springer.

BURKE, R., MOBASHER, B., WILLIAMS, C., AND BHAUMIK, R. (2006). Classification features for attack detection in collaborative recommender systems. In *ACMSIGKDD International Conference on Knowledge Discovery and Data Mining*. Pp. 542–547.

CANNY, J. F. (2002). Collaborative filtering with privacy via factor analysis. *ACM Conference on Research and Development in Information Retrieval (SIGIR)*. Pp. 238–245.

Charu C. Aggarwal and Philip S. Yu, (2008), "A General Survey of Privacy-Preserving Data Mining Models and Algorithms".

Chen, M., & Singh, J. P. (2001). Computing and using reputations for internet ratings. *Proceedings of the 3rd ACM Conference on Electronic Commerce* (pp. 154-162).

CHIRITA, P.-A., NEJDL, W., AND ZAMFIR, C. (2005). Preventing shilling attacks in online recommender systems. *ACM International Workshop on Web Information and Data Management (WIDM)*. Pp. 67–74.

Gayatri Nayak (2011), "A Survey on Privacy Preserving Data Mining: Approaches and Techniques".

Healthcare Success. (2014). *How Patients Judge Provider Quality*. [online] Available at: http://www.healthcaresuccess.com/blog/doctor-marketing/good-doctor-vs-bad-doctor-patients-judge-provider-quality.html [Accessed 10 Aug. 2016].

Herlocker, J., Konstan, J.., Terveen, L. and Riedl, J. (2004): Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst*. 22(1), 5–53.

Hoens, T.R (2010), Reliable medical recommendation systems with patient privacy. *Proceedings of the 1st ACM International Health Informatics Symposium*, pp. 173-182.

Hu, N., Zhang, J., & Pavlou, P. A. (2009). Overcoming the J-shaped distribution of product reviews. *Communications of the ACM, 52*(10), 144-147.

Institute of Medicine (IOM) (2001). Crossing the quality chasm: *A new health system for the 21st Century*, National Academies Press, Washington, D.C.

Jeckmans, A. J. P., Beye, M., Erkin, Z., Hartel, P., Lagendijk, R. L., & Tang, Q. (2013). Privacy in recommender systems. N. Ramzan, R. van Zwol, J.-S. Lee, K. Clüver, & X.-S. Hua (Eds.), *Social Media Retrieval*, 263–281. London, UK: Springer.

KATZENBEISSER, S. & PETKOVIC, M. (2008). Privacy-preserving recommendation systems for consumer healthcare services. *IEEE International Conference on Availability, Reliability and Security (ARES)*. Pp. 889–895.

Koren, Y., Bell and R., Volinsky, C (2009). Matrix factorization techniques for recommender systems. Computer, pp 30–37.

Ku, Y.-C., & Tai, Y.-M. (2013). What Happens When Recommendation System Meets Reputation System? The Impact of Recommendation Information on Purchase Intention. In System Sciences (HICSS), 2013 46th Hawaii International Conference on, 1376-1383, IEEE.

LAM., S. K. AND RIEDL, J. (2004). Shilling Recommender systems for fun and profit. *ACM International Conference on World Wide Web (WWW)*. Pp. 393–402

Mahesh Dhande, N.A.Nemade and Yogesh Kolhe, (2013), "Privacy Preserving in K-Anonymization Databases Using AES Technique".

Manish Sharma and Atul Chaudhary, (2013), "A Review Study on the Privacy Preserving Data Mining Techniques and Approaches".

Mbandu A.S et al. (2015), Recommender Privacy-Preserving Reputation Based Medical Services Scheme using a variant of ElGamal. *International Journal of Computer and Communication Engineering*. Pp. 235-245.

MEHTA, B., HOFMANN, T., AND FANKHAUSER, P. (2007). Lies and propaganda: detecting spam users in collaborative filtering. *International Conference on Intelligent User Interfaces*. Pp. 14–21.

MILLER, B., KONSTAN, J., AND RIEDL, J. (2004). Pocketlens: Toward a personal recommender system. *ACM Transactions on Information Systems 22,* 3, 437–476.

Mosadeghrad, A. (2012). A Conceptual Framework for Quality of Care. *Mat Soc Med*, 24(4), p.251.

Nabil A. M and Govardhan A. (2010) “A Comparison Between Five Models Of Software Engineering”, *IJCSI International Journal of Computer Science Issues*, 7. Pp 94-101.

POLAT, H. AND DU, W. (2005). SVD-based collaborative filtering with privacy. *ACMSymposium on Applied Computing (SAC)*. 791–795.

Rendle, S. et al. (2011). Fast context-aware recommendations with factorization machines. *International Conference on Research and Development in Information Retrieval*, pp. 635–644.

Resnick, P., & Zeckhauser, R. (2002). Trust among strangers in Internet transactions: Empirical analysis of eBay's reputation system. *Advances in Applied Microeconomics*, *11*, pp. 127-157.

Schafer, J.B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filtering recommender systems. In P. Brusilovsky, A. Kobsa, & W. Nejdl (Eds.), *The Adaptive Web*, LNCS 4321, 291–324. Berlin Heidelberg, Germany: Springer.

Sciencedirect.com. (2016).*Recommendation systems: Principles, methods and evaluation* . [online] Available at: http://www.sciencedirect.com/science/article/pii/S1110866515000341 [Accessed 4 Aug. 2016].

SRIVATSA, M., XIONG, L., AND LIU, L. (2005). Trustguard: countering vulnerabilities in reputation management for decentralized overlay networks. *International Conference on World Wide Web (WWW)*. Pp. 422–431.

Suchistra (2015), Techniques for Privacy Preservation in Data Mining. *International Journal of Engineering Research & Technology (IJERT).* Pp.490-493

Yu, P., Lin, L. and Yao, Y. (2016). A Novel Framework to Process the Quantity and Quality of User Behavior Data in Recommender Systems. *Web-Age Information Management*, pp.231-243.

Yu, P., Lin, L., Wang, F., Wang, J., Wang, M. (2014). Improving recommendations with collaborative factors. *Li, F., Li, G., Hwang, S.-W., Yao, B., Zhang, Z. (eds.) WAIM 2014. LNCS*, vol. 8485, pp. 30–33.

ZHAN, J. (2010). Privacy-preserving collaborative recommender systems. *Systems, Man, and Cybernetics, Part C: Applications and Reviews 40*, 4, pp. 472–476.

Zhuang, Y., Chin, W.S., Juan, Y.C., Lin, C.J. (2013). A fast parallel SGD for matrix factorization in shared memory systems. *International ACM Conference on Recommender Systems*, pp. 249–256.

APPENDIX

Appendix 1: Questionnaire

1) How do you Rate the overall recommendations made by the system?

- ☐ Very useful
- ☐ Useful
- ☐ A bit useful
- ☐ Not useful
- ☐ Not sure

2) How easy to use is the system?

- ☐ Very useful
- ☐ Useful
- ☐ A bit useful
- ☐ Not useful
- ☐ Not sure

3) Would you recommend the system to other healthcare users?

- ☐ Very useful
- ☐ Useful
- ☐ A bit useful
- ☐ Not sure

- [ ] Not sure

4) Would you use the system again/ another time to provide feedback or obtain recommendations?
   - [ ] Very useful
   - [ ] Useful
   - [ ] A bit useful
   - [ ] Not useful
   - [ ] Not sure

5) How relevant were the recommendation made for healthcare providers by the system?
   - [ ] Very useful
   - [ ] Useful
   - [ ] A bit useful
   - [ ] Not useful
   - [ ] Not sure

6) How relevant were the recommendation made for health conditions by the system?
   - [ ] Very useful
   - [ ] Useful
   - [ ] A bit useful
   - [ ] Not useful
   - [ ] Not sure

7) How useful was the feedback you submitted for providing recommendations?

☐ Very useful

☐ Useful

☐ A bit useful

☐ Not useful

☐ Not sure

8) What is your general perception about the system? Do you think that the system will help users in obtaining quality care from healthcare?

____

Appendix 2: Data collected during survey

		User review Frequency for each review Question				
		0	**1**	**2**	**3**	**4**
Review Questions	How do you Rate The overall recommendations made by the system?	0	4	6	32	8
	How easy to use is the system?	2	2	8	18	20
	Would you recommend the system to other healthcare users?	2	5	6	14	23
	Would you use the system again/ another time to provide feedback or obtain recommendations?	1	4	8	16	21
	How relevant were the recommendation made for healthcare providers by the system?	0	2	11	13	24
	How relevant were the recommendation made for health conditions by the system?	0	2	10	15	23
	How useful was the feedback you submitted for providing recommendations?	1	4	9	17	19

LEGEND

0= Not sure

1= Not useful

2= A bit useful

3= Useful

4= Very useful

Figure 23: Survey data

Appendix 3: Sample code

List of Health providers and health conditions

```
<?php
                    $attributes = array('class' => 'form-horizontal','method' => 'post', 'id' => 'frm_buyairtime',
'name' => 'frm_buyairtime');
                    $submiturl = LAYOUT_URL . 'index.php/index/index';
                    echo form_open_multipart($submiturl, $attributes);
                ?>
                    <div class="box box-solid">
                        <div class="box-header with-border">
                            <h3 class="box-title">Select the Hostipal and Disease from the list</h3>
                        </div><!-- /.box-header -->
                        <div class="box-body">
                            <div class="row">
                                <div class="col-md-6" style="padding:0 50px">
                                    <div class="form-group">
                                        <label>Select Hospital</label>
                                        <select name="hospital" class="form-control">
                                            <option value="1">NAIROBI HOSPITAL</option>
                                            <option value="2">MATER HOSPITAL</option>
                                            <option value="3">KENYATTA NATIONAL HOSPITAL</option>
                                            <option value="4">THE AGA KHAN HOSPITAL</option>
                                            <option value="5">KAREN HOSPITAL</option>
                                            <option value="6">JAMAA HOSPITAL</option>
                                            <option value="7">AVENUE HOSPITAL</option>
                                            <option value="8">GURU NANAK HOSPITAL</option>
                                            <option value="9">GETRUDE GARDEN CHILDRENS HOSPITAL</option>
                                            <option value="10">MP SHAH HOSPITAL</option>
                                        </select>
                                    </div>
                                </div>

                                <div class="col-md-6" style="padding:0 50px">
                                    <div class="form-group">
                                        <label>Select Condition</label>
                                        <select name="condition" class="form-control">
                                            <option value="1">DIABETIES</option>
                                            <option value="2">DEPRESSION</option>
                                            <option value="3">KIDNEY FAILURE</option>
                                            <option value="4">DENTISTRY</option>
                                            <option value="5">BREAST CANCER</option>
```

```
                    <option value="6">BONE TUMOR</option>
                    <option value="7">EYE HEALTH</option>
                    <option value="8">HEART DISEASE</option>
                    <option value="9">LEUKEMIA</option>
                    <option value="10">PNEUMONIA</option>
                  </select>
                </div>
              </div>
            </div>
          </div><!-- /.box-body -->
        </div><!-- /.box -->
```

Healthcare quality metrics (User feedback/ratings)

```
        <div class="box box-solid">
          <div class="box-header with-border">
            <h3 class="box-title">How would you rate service by the Doctor/Physician in
charge?</h3>
          </div><!-- /.box-header -->
          <div class="box-body">
            <table class="table table-bordered">
              <tr>
                <th style="width: 10px">#</th>
                <th style="width: 50%">Question</th>
                <th style="width: 10px">Excelent</th>
                <th style="width: 10px">Good</th>
                <th style="width: 10px">Average</th>
                <th style="width: 10px">Poor</th>
                <th style="width: 10px">Terible</th>
              </tr>
              <tr>
                <td>1</td>
                <td>Promptness in attendance</td>
                <td><input type="radio" name="doctor_promptness_attendance"
id="optionsRadios2" value="5" required></td>
                <td><input type="radio" name="doctor_promptness_attendance"
id="optionsRadios2" value="4"></td>
                <td><input type="radio" name="doctor_promptness_attendance"
id="optionsRadios2" value="3"></td>
                <td><input type="radio" name="doctor_promptness_attendance"
id="optionsRadios2" value="2"></td>
                <td><input type="radio" name="doctor_promptness_attendance"
id="optionsRadios2" value="1"></td>
              </tr>
```

```
                    <tr>
                      <td>2</td>
                      <td>Listens and Attentiveness</td>
                      <td><input type="radio" name="doctor_listens_attentive" id="optionsRadios2"
value="5" required></td>
                      <td><input type="radio" name="doctor_listens_attentive" id="optionsRadios2"
value="4"></td>
                      <td><input type="radio" name="doctor_listens_attentive" id="optionsRadios2"
value="3"></td>
                      <td><input type="radio" name="doctor_listens_attentive" id="optionsRadios2"
value="2"></td>
                      <td><input type="radio" name="doctor_listens_attentive" id="optionsRadios2"
value="1"></td>
                    </tr>
                    <tr>
                      <td>3</td>
                      <td>Proper Explantion of condition and treatment</td>
                      <td><input type="radio" name="doctor_proper_explantions" id="optionsRadios2"
value="5" required></td>
                      <td><input type="radio" name="doctor_proper_explantions" id="optionsRadios2"
value="4"></td>
                      <td><input type="radio" name="doctor_proper_explantions" id="optionsRadios2"
value="3"></td>
                      <td><input type="radio" name="doctor_proper_explantions" id="optionsRadios2"
value="2"></td>
                      <td><input type="radio" name="doctor_proper_explantions" id="optionsRadios2"
value="1"></td>
                    </tr>
                    <tr>
                      <td>4</td>
                      <td>Care and concern</td>
                      <td><input type="radio" name="doctor_care_concern" id="optionsRadios2"
value="5" required></td>
                      <td><input type="radio" name="doctor_care_concern" id="optionsRadios2"
value="4"></td>
                      <td><input type="radio" name="doctor_care_concern" id="optionsRadios2"
value="3"></td>
                      <td><input type="radio" name="doctor_care_concern" id="optionsRadios2"
value="2"></td>
                      <td><input type="radio" name="doctor_care_concern" id="optionsRadios2"
value="1"></td>
                    </tr>
                    <tr>
                      <td>5</td>
```

```
                    <td>Professional and Courteous </td>
                    <td><input type="radio" name="doctor_professional_courteous"
id="optionsRadios2" value="5" required></td>
                    <td><input type="radio" name="doctor_professional_courteous"
id="optionsRadios2" value="4"></td>
                    <td><input type="radio" name="doctor_professional_courteous"
id="optionsRadios2" value="3"></td>
                    <td><input type="radio" name="doctor_professional_courteous"
id="optionsRadios2" value="2"></td>
                    <td><input type="radio" name="doctor_professional_courteous"
id="optionsRadios2" value="1"></td>
                  </tr>
                </table>
              </div><!-- /.box-body -->
            </div><!-- /.box -->

            <div class="box box-solid">
              <div class="box-header with-border">
                <h3 class="box-title">How would you rate service at the reception or front-
office?</h3>
              </div><!-- /.box-header -->
              <div class="box-body">
                <table class="table table-bordered">
                  <tr>
                    <th style="width: 10px">#</th>
                    <th style="width: 50%">Question</th>
                    <th style="width: 10px">Excelent</th>
                    <th style="width: 10px">Good</th>
                    <th style="width: 10px">Average</th>
                    <th style="width: 10px">Poor</th>
                    <th style="width: 10px">Terible</th>
                  </tr>
                  <tr>
                    <td>1</td>
                    <td>Promptness/ Responsiveness/ Attentive to queries</td>
                    <td><input type="radio" name="reception_promptness_attendance"
id="optionsRadios2" value="5" required></td>
                    <td><input type="radio" name="reception_promptness_attendance"
id="optionsRadios2" value="4"></td>
                    <td><input type="radio" name="reception_promptness_attendance"
id="optionsRadios2" value="3"></td>
                    <td><input type="radio" name="reception_promptness_attendance"
id="optionsRadios2" value="2"></td>
```

```
                  <td><input type="radio" name="reception_promptness_attendance"
id="optionsRadios2" value="1"></td>
                </tr>
                <tr>
                  <td>2</td>
                  <td>Listens and Attentiveness</td>
                  <td><input type="radio" name="reception_polite_friendly" id="optionsRadios2"
value="5" required></td>
                  <td><input type="radio" name="reception_polite_friendly" id="optionsRadios2"
value="4"></td>
                  <td><input type="radio" name="reception_polite_friendly" id="optionsRadios2"
value="3"></td>
                  <td><input type="radio" name="reception_polite_friendly" id="optionsRadios2"
value="2"></td>
                  <td><input type="radio" name="reception_polite_friendly" id="optionsRadios2"
value="1"></td>
                </tr>
                <tr>
                  <td>3</td>
                  <td>Waiting time to see doctor/ access service</td>
                  <td><input type="radio" name="reception_wait_service" id="optionsRadios2"
value="5" required></td>
                  <td><input type="radio" name="reception_wait_service" id="optionsRadios2"
value="4"></td>
                  <td><input type="radio" name="reception_wait_service" id="optionsRadios2"
value="3"></td>
                  <td><input type="radio" name="reception_wait_service" id="optionsRadios2"
value="2"></td>
                  <td><input type="radio" name="reception_wait_service" id="optionsRadios2"
value="1"></td>
                </tr>
              </table>
            </div><!-- /.box-body -->
          </div><!-- /.box -->

          <div class="box box-solid">
            <div class="box-header with-border">
              <h3 class="box-title">How would you rate our office environment</h3>
            </div><!-- /.box-header -->
            <div class="box-body">
              <table class="table table-bordered">
                <tr>
                  <th style="width: 10px">#</th>
                  <th style="width: 50%">Question</th>
```

```
                    <th style="width: 10px">Excelent</th>
                    <th style="width: 10px">Good</th>
                    <th style="width: 10px">Average</th>
                    <th style="width: 10px">Poor</th>
                    <th style="width: 10px">Terible</th>
                  </tr>
                  <tr>
                    <td>1</td>
                    <td>Cleaniness of hospital facilities/ General cleaniness</td>
                    <td><input type="radio" name="cleanliness_facilitiies" id="optionsRadios2"
value="5" required></td>
                    <td><input type="radio" name="cleanliness_facilitiies" id="optionsRadios2"
value="4"></td>
                    <td><input type="radio" name="cleanliness_facilitiies" id="optionsRadios2"
value="3"></td>
                    <td><input type="radio" name="cleanliness_facilitiies" id="optionsRadios2"
value="2"></td>
                    <td><input type="radio" name="cleanliness_facilitiies" id="optionsRadios2"
value="1"></td>
                  </tr>
                  <tr>
                    <td>2</td>
                    <td>Washrooms cleaniness</td>
                    <td><input type="radio" name="cleaniness_washrooms" id="optionsRadios2"
value="5" required></td>
                    <td><input type="radio" name="cleaniness_washrooms" id="optionsRadios2"
value="4"></td>
                    <td><input type="radio" name="cleaniness_washrooms" id="optionsRadios2"
value="3"></td>
                    <td><input type="radio" name="cleaniness_washrooms" id="optionsRadios2"
value="2"></td>
                    <td><input type="radio" name="cleaniness_washrooms" id="optionsRadios2"
value="1"></td>
                  </tr>
                </table>
              </div><!-- /.box-body -->
            </div><!-- /.box -->

            <div class="box box-solid">
              <div class="box-header with-border">
                <h3 class="box-title">Enter your email address</h3>
              </div><!-- /.box-header -->
              <div class="box-body">
                <div class="row">
```

```
                    <div class="col-md-6">
                        <div class="input-group">
                            <span class="input-group-addon"><i class="fa fa-envelope"></i></span>
                            <input type="email" name="email" class="form-control" placeholder="Email
Address">
                        </div>
                    </div>
                </div>
            </div><!-- /.box-body -->

            <div class="box-footer">
                <div class="row">
                    <div class="col-md-6">
                        <button type="submit" class="btn btn-block btn-success">SUBMIT YOUR THE
DETAILS</button>
                    </div>
                </div>
            </div>
        </div>
    </form>
```