MANAGING SECURITY RISK OF HEALTHCARE WEB APPLICATION: A DESIGN PERSPECTIVE

Thesis submitted in fulfilment of the requirement for

the degree of

Doctor of Philosophy

IN

INFORMATION TECHNOLOGY

Submitted by

Syed Anas Ansar

Supervised by

Prof.Raees Ahmad Khan

Co - Supervised by

Dr. AmitabhaYadav

Submitted to

DEPARTMENT OF INFORMATION TECHNOLOGY

BABASAHEB BHIMRAO AMBEDKAR UNIVERSITY

(A CENTRAL UNIVERSITY)

VIDYA VIHAR, RAEBARELI ROAD,

LUCKNOW – 226025, UTTAR PRADESH, INDIA

AUGUST 2021

DECLARATION

I, **Syed Anas Ansar**, solemnly declare that this thesis of research on "**Managing Security Risk of Healthcare Web Application: A Design Perspective**" is my original work. The study has been conducted under the guidance of **Prof. Raees Ahmad Khan**, at Department of Information Technology, Babasaheb Bhimrao Ambedkar University (A Central University), Lucknow, India, and co-guidance of **Dr. Amitabha Yadav,** Department of Software, Deen Dayal Upadhyay Kaushal Kendra, National Post Graduate College, Lucknow, India. It is further declared that to the best of my knowledge and belief it has not been submitted earlier for the award of any degree. I also undertake that the thesis is essentially free from all kinds of plagiarism.

Dated: 12/08/2021

(Syed Anas Ansar)
Researcher
Department of Information Technology
Babasaheb Bhimrao Ambedkar University
(A Central University)
Lucknow, Uttar Pradesh, India

CERTIFICATE

This is to certify that the thesis entitled "**Managing Security Risk of Healthcare Web Application: A Design Perspective**" submitted by **Mr. Syed Anas Ansar** is an original research work and has not been previously submitted in part or full for the award of any other degree or diploma to this or any other University.

This thesis submitted to Babasaheb Bhimrao Ambedkar University, Lucknow, satisfies all the requirements as stipulated in the Doctor of Philosophy (Ph.D.) regulations-1999 as amended in 2008/2010/2013 and it is fit for submission and evaluation for the award of the degree of Doctor of Philosophy of the University.

Co-Supervisor

Supervisor

Dated: 12 · Aug · 2021

Head of the Department

HEAD
Department of Information Technology
School For Information Science &
Babasaheb Bhimrao Ambedkar University
Lucknow

ACKNOWLEDGEMENT

The writing of this thesis has been an incredible journey and a monumental milestone in my academic life. I could not have embarked on this expedition and travelled this far without the passionate and continued support of supervisors, colleagues, friends, and family. Working as a Ph.D. scholar at Babasaheb Bhimrao Ambedkar University was a magnificent as well as challenging experience to me. In all these years, many people were instrumental directly or indirectly in shaping up my academic career. It was hardly possible for me to thrive in my research work without the precious support of these people. Here is a small gratitude to all those people.

I am indeed indebted to my distinguished guide and supervisor **Prof. Raees Ahmad Khan** for all his help during the course of the study. Certainly, I am short of words to convey my real feelings for his invaluable help and concern together with 'scholarly' insightful and 'critical' guidance. I do not hesitate to state that without his help it would not have been possible for me to complete the research work. I am also very grateful to my co-supervisor **Dr. Amitabha Yadav** for his guidance, support and consultations during the course of the study.

I must thank to **Prof. Raees Ahmad Khan** because despite of his tight time-schedule, he was always available to me all time to answer my queries, discussion of matters, helped me to learn from my mistakes, without even forcing his opinion on me. I spent countless hours with him for discussing research, writing research papers, proof reading of research manuscripts, making this thesis and talking about my life in general. Moreover, being the Head of the Department, Prof. Khan gave me the opportunity to pursue research work at the Department of Information Technology, Babasaheb Bhimrao Ambedkar University, Lucknow. I

received lots of motivation, encouragement, and support from him during my entire duration.

I feel deep sense of gratitude for my parents, especially for my father **Syed Ansar Ahmad** and mother **Syeda Nargis Ansar**, who formed a base of my vision and taught me the good things that really matter in my life. I am also grateful to my younger brother **Syed Aquib Ansar** and my younger sister **Syeda Rahmeena Ansar** because I couldn't have done it without them. In addition, I am also thankful to **Dr. Mohd. Waris Khan** (Assistant Professor) for his continuous encouragement, guidance, moral support, and consultations during the course of the study. I am thankful to SAQ Infosys for granting me the permission and needed assistance for data collection and experiments. I am also thankful to all my friends and colleagues for providing encouragement and support, especially Jaya Yadav, Kamran Suhail, and Manish Joshi. I express my sincere thanks to all the faculty members and office staff of the department for their time-to-time continuous encouragement and support. I express my sincere thanks to all the experts from India and abroad for providing me with their valuable observations during the Expert Opinions and peer review of my research papers. In summary, I would like to thank everyone for putting up with me for the entire period of my research work.

Syed Anas Ansar

ABSTRACT

The modern world is critically reliant on a broad range of software applications. Dependency on software applications is so high that life cannot be imagined without them. Information, no matter to which part of the globe it belongs, is available with a click of the mouse. Intensive security-oriented services ranging from internet banking, trading to online, buying and selling, booking an appointment to a doctor etc., are carried out unhesitatingly. These services require the privacy of the information and asset. When security intensive information is floating everywhere, anyone having malicious intent can misuse the information. This may harm an organization or individuals. Since decades, efforts are being made to estimate security risk in order to increase accountability, demonstrate compliance, and determine whether and by how much our investments in the product make our systems more secure.

Furthermore, the health sector is one of the most prime sectors where all the hi-tech applications are used. In this sector, medical personnel are entrusted with a vast number of responsibilities, and dealing with them is a more sophisticated as well challenging task. The healthcare sector has been linked to the technological world in order to ease the responsibilities as well as workloads of the healthcare staff. This was made possible by integrating IT (Information Technology) into the healthcare sector. Apart from these technological advancements, several statistics have demonstrated data breaches instances that have affected both, i.e., patients and Healthcare Information systems. Thousands of healthcare records can be compromised by security breaches.

In addition, to secure an individual's as well as HIS's data, three major security factors and privacy goals are needed, which is commonly known as the CIA triad. The significant necessity of the CIA trio is;

Confidentiality must be included for highly sensitive data; integrity is important because it may be fatal to provide an inaccurate procedure based on faulty data of medical, and availability is as necessary because the data must be available on time for adequate treatment.

In the healthcare web application, the privacy of individual and organizational data is extremely important, and currently it has become a major challenge to shield healthcare information. The major challenge introduced in the healthcare web application is due to huge data growth. Furthermore, COVID-19 (i.e., the current pandemic situation) has resulted in an unexpected spike in healthcare data, which has impacted both the healthcare web applications and hospitals. Managing these healthcare data and securing it from intruders has become a complex task for security experts. Nowadays, the main objective of the researchers and security experts is to minimize security vulnerabilities in the healthcare web application by mitigating and assessing the security risk factors. So, some dedicated steps are required to enhance the security of healthcare web applications, which may help in securing and protecting them in order to ensure transparency and assess security risk. This is why security professionals prefer to take a step up on the design phase to reduce security risks. It will assist in designing secure web applications in the healthcare sector. Furthermore, it may also assist in overcoming from threats and protecting it from cyber-attacks by early detection and mitigation of security risk factors in the design phase.

In order to gain a competitive edge, developers and researchers need to create a viable security risk assessment framework so as to minimize critical healthcare web application failures. Though it is highly difficult to create a perfectly secure healthcare web application system, but one can surely reduce the security risks by following a fool-proof and meticulously designed strategy with the inclusion of security attributes. In

addition to this, the researcher has made an effort to overcome this issue and proposed a framework to assess security risk of the healthcare web application. This framework incorporates five phases, including Factors Identification, Mapping, Assessment, Statistical Analysis and Review and Revision.

The first phase, i.e., Factors Identification, in which the identification and selection of security risk factors as well as their corresponding security attributes have been made on the basis of a comprehensive literature review and expert's opinions. The relationship among security attributes and security risk factors has been developed in the second phase. In addition, an integrated Fuzzy AHP-TOPSIS approach is used for security risk assessment in the third phase. Where Fuzzy AHP is used to prioritize security risk factors, and the impact of security attributes on various alternatives is calculated with the help of Fuzzy AHP-TOPSIS. Furthermore, sensitivity analysis and empirical validation are carried out in the second last phase of the framework. In the last phase, review and revision will be undertaken only when required, which facilitates a retrospect of the entire development activity and aid in making changes whenever necessary.

It is apparent from the validation of the proposed framework that it may be significantly helpful to keep in check the potential risk and vulnerabilities from the early design phase till the end. The proposed framework has shown satisfactory results with respect to other mentioned approaches. It may also form the basis for the development of new modified or refined approaches. Like any other research, the current work may also suffer from certain limitations, therefore to achieve a generalized result and implementation of the proposed model, further study may be conducted on a large application.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1
INTRODUCTION

1.1 Background

The world has witnessed an enormous rise in software industries from last decades. Nowadays, technology has made complicated work much more accessible and more straightforward. Nearly most of the services are available through computers, from browsing the web to scheduling an appointment [1]. The numerous tasks are handled and managed through different software and applications, which has become a critical element of our daily lives. As human activities are heavily reliant on software, hence any untidiness during the development of software may lead to the life and death of human beings [2].

The advancement of technology enables individuals and organizations to process, create, store, and exchange vast quantities of data from anywhere at any time [3]. Almost every area has seen massive growth in software deployment. With the ease of its facilities, there is fear too, because news headlines are continuously threatening about the data breach, which indicates an immediate need to tackle these issues [4]. This is why software security has become the focus of rigorous studies and a key component for steady stream aspects that strengthened software engineer's ability to protect applications and build secured software [5].

An enormous amount of information for secure development is available in books, open literature, or on the internet etc.; despite all these, attackers always think ahead of it and find novel ways to exploit vulnerabilities. Some researchers have also mentioned that security is not taken into account as a priority by developers in the initial phase and typically think that security is a post-development practice. In most

development organizations, security is implemented by developers or software engineers as a fix after development of application, but security is not an afterthought, rather it is an evolving feature of software [6].

Researchers and experts realize that software security should not be reflected as an added value or need for gold plating [7]. For ensuring the security of software, the best risk management practices should be applied from the initial phase and should be spread across the entire process [8]. Therefore, when highly secured applications are developed, it needs to involve more sophisticated security operations [9]. In this study, the researcher has focused on the security-related problem in software development. The main objective is to provide an insight into security risk for the sole purpose of creating secure and robust software from the initial phase, i.e., the design phase. The regularization of software security maintains the whole system's consistency [10, 11].

The health sector is one of the most important sectors in which all the hi-tech applications are used in medical institutions. Here, vast numbers of responsibilities are allotted to medical staff, and dealing with it is a more sophisticated and challenging job. There are lots of records of patients that have to be managed, including tracking inventory, working schedules of doctors, records for keeping bills, patient’s reports, etc. [12]. To ease all these responsibilities and workloads, the medical system has been linked to the technical world so that the workload of the medical staff can be made easier. All this was possible only with the integration of information technology to our medical system, and HMS (Hospital Management System) is the result. An efficient HMS can share useful knowledge rather than just storing and simply presenting data. In addition, it maximizes employee usage as well as helps in clinical decisions. It also provides automatic services like informing important tasks to various software users and movements of employees etc. [13].

HMS manages the administration of the hospital (i.e., clinical and financial aspects) by processing the hospital database and provide information in the correct form at the right time, and right place, by minimizing human efforts to take decisions efficiently. The primary goal of HMS is to provide efficient and effective service by eliminating the manual procedure and boosting the efficiency of medical institutions in terms of effective and rapid healthcare [14].

There has been a lot of advancement in the medical institutions that changes the working style of hospital staff due to the ICT (Information Communication Technology) in healthcare sectors. The ICT has enhanced the system's capacity to gather, store, and transmit information, it improves the working system of hospitals but may create severe problems for patients. This progress in hospital systems made the work of the hospital staff very convenient and straightforward rather than manual work, but security has been a very complicated issue despite all these advances. It poses many specific issues; threats in computers and networks can result in disclosure of individual information, reputational harm, loss of money, and legal consequences. Despite of warnings from a wealth of public breach notices, numerous of hospitals are insufficiently prefabricated to deal with today's computer-based attacks [15]. To learn the security issues of HMS, one need to understand the risks of healthcare and its consequences and how the information is used in healthcare.

In 1976, Professor Alan F. Westin elaborated a healthcare information flow, i.e., how information is used in this sector as well as the healthcare information is used in different sectors from Direct Patient Care to Support Activities, Support Activities to Commercial and also for social uses [16]. The HMS contains data in the electronic format, i.e., e-data, which collects and stores patient's medical data that can be accessed from any location and provides centralized data [17]. This e-record

contains a lot of ordinary and common information about the patient in the medical records, such as blood pressures report and any injury. In addition, it may contain sensitive and confidential information. The new advances lead to a state where leakage of the information in medical records of patients pose new threats to privacy and create a security risk [18]. It is important to monitor and control its accessibility because it can harm the patient [19]. These e-records of patients are accessed by enterprises that maintain such databases to offer EHR (Electronic Health Record) services. The patient's data serves as a key point for the collection, storing, and dissemination of individual data to third parties mostly provided to service providers of healthcare, medical or drug stores, and among others hospitals doctors.

The HMS may control the disclosure of information inside the medical institutions. It may prevent accidental disclosures, insider curiosity of medical staff- to access the report of celebrities or their colleague's medical reports, insider subornation-release patients report to an outsider for their profit or revenge and may control third party access or secondary usage. But in case/s of unauthorized access by exploiting the vulnerability of software, it would be difficult for hospital management to handle it on time. Because of the deadlines, developers build software within the specified duration that may cause loopholes in the software. The most contentious concern in recent years is how technology jeopardizes the privacy of patient's data [19].

Apart from this progress in technology, numerous statistics have shown data breach instances that have affected both the patients and HMSs in the healthcare field. Thousands of health records can be compromised by security breaches. To secure an individual's data, three major security factors and privacy goals are commonly identified and known as CIA (Confidentiality, Integrity, and Availability) triad [17,20].

There is a significant necessity of the CIA triad; confidentiality must be included for highly sensitive data; integrity is important because it may be fatal to provide an inaccurate procedure based on faulty data of medical, and availability is as necessary because the data must be available on time for adequate treatment. In the medical field, the security and privacy of individual data are critical, and it is a major challenge to shield the healthcare data [21]. In the healthcare sector, several challenges are introduced in HMS like huge data growth.

In addition, after the covid-19 pandemic situation that has created a sudden spike in health data, hospitals as well as healthcare have been impacted as well. Managing this data and protecting it from attackers has become difficult for security experts. Presently, the major objective of security professionals and researchers is to minimize the security threats by mitigating and evaluating the risk factors in the web applications of healthcare [22].

In the healthcare sector, for enhancing the security mitigating the risks, some dedicated steps are needed. It may help to secure and protect healthcare web applications for ensuring transparency and evaluating security. This is the reason; developers prefer to take a step up on the design phase to reduce vulnerability. It will help to build secure web applications in the healthcare sector. It may help to overcome the threats and protect from cyber-attacks by early detection and mitigation of risk factors in the design phase.

Further, from the study of available statistics, it is observed that security plays a crucial role in the success and failure of healthcare web application systems. Therefore, from the detailed study of healthcare web applications used in all types of fields, it can be observed that a healthcare web application is ubiquitous to both (i.e., personal and

professional life) and it is a source of vital information. Hence, its security is of utmost concern.

Researchers and developers must design a feasible security risk assessment framework in order to minimize critical healthcare web application failures. Though creating a perfectly a secure healthcare web application is extremely tough task, but one can reduce security risks by implementing a fool-proof and precisely designed approach that includes security attributes. Techniques of security optimization will further help the security practitioners as well as researchers to reduce the time and cost required for the development of a healthcare web application system. Also, in-depth identification, analysis and mitigation will deliver a quality product.

1.2 Hospital Management System

The committee of WHO, in 1956 defined Hospital as "A vital part of a medical and social institution whose role is to accommodate maximum healthcare services to their community from cure to prevention; the hospitals are also serves as a centre for the training of health workers as well as bio-social research" [23]. The healthcare centres are an integral part of human lives, providing appropriate services and the cure to citizens who suffered from different illnesses due to environmental conditions, mental stress, or other reasons. Hospitals need to keep track of their everyday activities to deal with the day-to-day workload. Different operational tasks are performed, which include patient's records, nurses and ward boy's schedules, generating bills, recording patient immunization, patient diagnosis information, managing appointments of doctors and other running staff who coordinate with each other to run hospitals efficiently and effectively [24].

It is very tedious and error-prone to deal with all such activities and to keep track of all the records and tasks on paper. This procedure creates complexities, redundancy, and inconsistency, making the system very slow and time-consuming; that's why hospital management is the most complex procedure among all administrative organisations. To handle these manual works, an automated system is implemented by using a web application called Hospital Management System (HMS) to deal with all such activities and maximizing the optimization level.

HMS is a web-based computerized application used for storing data, managing records, tracking, and monitoring prescriptions. It can manage huge records of patients at a time and controls multiple works. It ensures that all users work as per their roles assigned by the system and can effectively complete their tasks on time. A successful software enables the system to insert and edit data easily; it summarises the outcomes, quickly update and rectify the errors at the same time [25]. Let's look at the following illustration for a detailed overview of HMS in figure 1.1, which describes the function of the hospital operated by HMS.

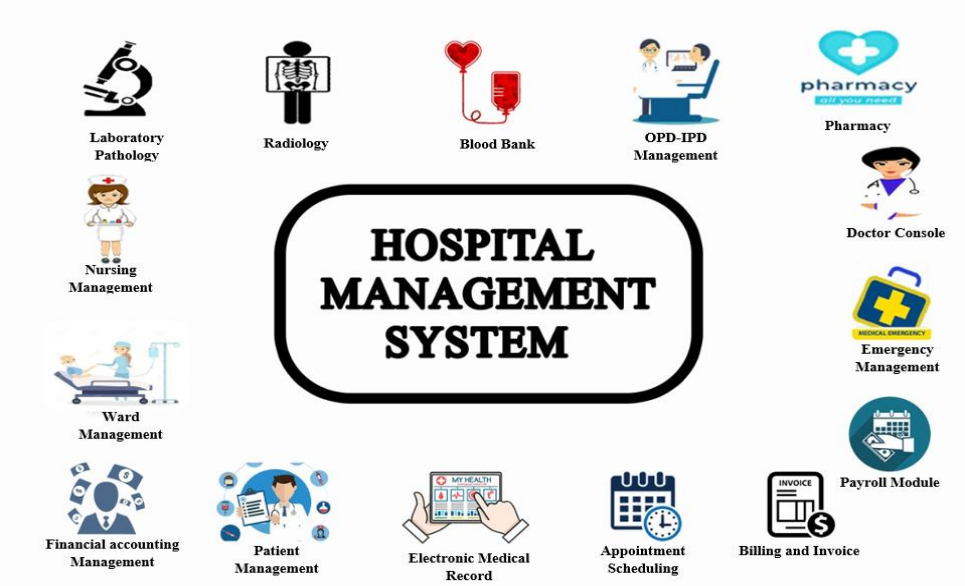

Figure 1.1: Hospital Management System

The web application of hospital deals with a number of tasks, which depends on various features of the software that manages several tasks of the hospital system. Few of the functionality of HMS is mentioned in figure 1.1, and these functionalities are managed by three user groups: hospital management/staff, patients, and third-parties [26]. Hospital management and staff take charge of several jobs include patient registration, help desk, OPD-IPD management, billing, invoicing, whereas patients can book appointments, pay bills, third parties provide insurance and companies supply drugs. In OPD, patients come for their check-up who suffered from health issues whereas; in IPD patients are admitted for their treatment of the disease. Pharmacy keeps records of medicine sales, current stock details, and bills of medicine, it provides general overflow.

Pathology and Radiology keep and maintain the test reports of patients. Blood banks are always updated with status, issue details, and donor. EMR (Electronic Medical Records) are uploaded on the software application to be shared with medical practitioners from different hospital departments with relevant details, such as medical tests, diagnoses, history of care, test results, and so on. The module of patient management acquires information of the patient from intake (for the diagnosis or treatment of the patient) to discharge. The work of cash or bank, payments receipt, ledger entry, profit and loss, the balance sheet of the hospital is made through the module of financial accounting, and payroll modules deal with salary slips printing, PF statements, salary certificates, etc. Nursing supervises, staff of nurses in the hospital, manage the routine tasks of patients to improve health conditions and cure them [27].

Hospital management deals with ward allotment, vacant OT (Operation Theatre) number, doctors and staff appointments admit and discharge details, emergency management. More and more jobs are done

in a single place and are managed by HMS software [28]. The workflow of HMS is shown in figure 1.2.

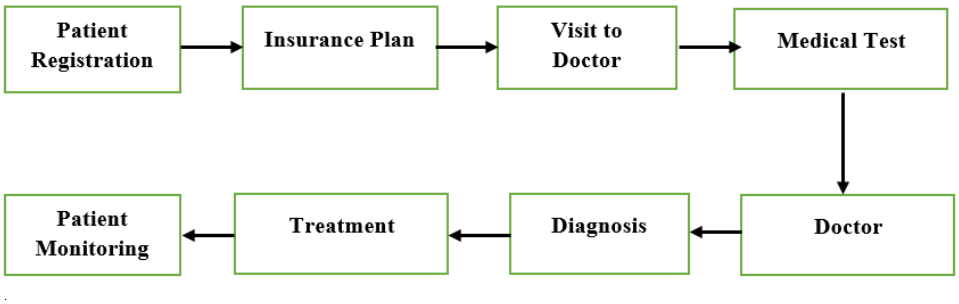

Figure 1.2: Workflow of Hospital Management System

HMS facilitates users to access data quickly as per the access control facility from the healthcare database as easily as one can insert and searches the records by authenticating the users and prevent unauthorized access. The HMS user has to follow figure 1.2 to access the HMS facility [28]. The patients can book their appointment by registering on the HMS application that synchronizes with insurance providers; those interested should approve it, and the patient appointment with the doctor is set. All these processes are conducted via the web application of HMS.

Patients have to visit for medical or diagnostic tests, as per the doctor's recommendation. These medical tests may be used for research purposes or study purposes to obtain accurate information about specific illnesses or diseases and also to track the care of the patient. These health records are preserved in digital or electronic format, ensuring an automated patient database. The healthcare web application introduces the concept of a centralized database that eradicates redundancy. It shortens the amount of time, especially when data is accessed and retrieved from the database to search previous records of patients. This software increases the accuracy and enhances the reliability of healthcare records

and provides effective storage and usage by electronic processing of data. Such technological advances have brought many changes in the working style of the medical staff, and the progress of the health departments made individuals more dependent on technology. In each and every field of medical, performance and interactivity should be achieved by ensuring security [29]. It provides confidentiality as well as integrity to sensitive data and can be accessed at any time from any location. These all can be done by implementing the CIA triad. The security-related flaw in the software makes it vulnerable and introduces risk for the information processed through it.

1.3 Healthcare Application Security

Hackers, viruses, and worms can be a serious threat to the security as well as privacy of healthcare web applications. In recent years, several incidents of accidental loss or theft of sensitive clinical data have been reported [30-33]. According to a recent study, every year almost 25 million compelled authorizations for the disclosure of healthcare records in the United States [33]. Security estimation of healthcare web applications focuses on the functional aspects of the application and its ability to endure a malicious attack and recover without data loss or any other abnormality [32]. The security estimation process assures the practitioners of the healthcare web application that it will not deteriorate by the presence of any kind of vulnerabilities and will always be able to use mitigation techniques in any situation. This process maximizes the success of user satisfaction on the healthcare web application system.

In the current times, security issues are constantly evolving due to the heterogeneous nature of healthcare web applications. Security estimation of healthcare web application is imperative for making it reliable and secure. It reveals those vulnerabilities that can affect healthcare web application's integrity, violates confidentiality/privacy

norms, and exploits loopholes in the design. These vulnerabilities may result in exhaustive data theft, malware, spyware injection and may cause failure in the entire healthcare web application. As discussed in the Introduction section, it can be contemplated that severe security flaws of healthcare web applications can steamroll into total application failure, culminating in substantial financial and physical losses [34].

It should be understood that security is not a product; rather, it is a process, and it should be taken into account from beginning to the end of healthcare web application development life cycle [32]. It cannot be incorporated or added as an additional feature at the end of the development life cycle or after the deployment as an afterthought. Consumers demand feature-rich and sophisticated healthcare web applications within no time. This compels the practitioners to produce huge and poorly-written healthcare web applications and release them without subjecting it to security estimation. For analysis, it is hard to contemplate and device exhaustive anti-security inputs and a challenging task to indorse the healthcare web application system's endurance and reliability against infinite numbers of security threats [35]. Healthcare web application’s security estimation is a quality assurance process through which a developer can understand the user's needs, design and analyse it thoroughly during the development life cycle before delivering it to the user. According to the report, there are infinite numbers of ways an attacker can disable or handicap healthcare web application system. However, no technique effectively covers all the security estimation features. Therefore, a balanced approach is required incorporating all the security attributes to prepare the analysis plan and plant it in the design and prepare a framework for the healthcare web application.

1.3.1 Healthcare Application Confidentiality

The communications in healthcare web applications are mainly in electronic form. This can lead to numerous threats to systems stability. Medical confidentiality is a collection of rules that restrict access to an unauthorised user or unauthorized disclosure of data, i.e., information exchange between individuals and healthcare professionals [34]. It is a higher measure of security policy that is undertaken to protect the database and to prevent the access of unauthorized users [36]. It ensures a certain degree of privilege to an authorized person who is permitted to access, make changes and download information. This privilege is restricted or not granted to the unauthorized users/personnel [37]. Confidentiality is already described as a pillar of healthcare ethics since from Hippocrates [38]. It is a broad security concept implemented at all stages of a healthcare web application system such as processing, storage, retrieval and display of information. It is ensured that the data is stored in an encrypted format and only the authorized user is permitted to decrypt and access the information. In addition, it also checks that the data is not stored in a plain format, so that leakage or misappropriation of the data or violation of the security parameters could not be commenced [39].

Hence, it strengthens a trusted binding mechanism of design and all its components, assuring that the sanctity of data is preserved and not violated by any intruders. During the design phase, all the components are bound together through coupling to create a mechanism that could protect the data from being violated or tampered by intruders [40]. Confidentiality runs deep down into the multi-faceted structure of healthcare web application systems to establish a security apparatus. Through which one can keep an eye on unauthorized attempts to gain access into the restricted or encrypted areas inside a healthcare system [41].

1.3.2 Healthcare Application Integrity

Integrity can be defined as the accurateness of data/information at storage or during transmission [34]. When it comes to managing hospital documents and healthcare web application's data, the value of data integrity is extremely high. A strong data integrity plan holds healthcare web applications and health services in line and guarantees patient confidence. Compromised data on healthcare may lead to misdiagnosis or inaccurate treatment and without proper permission; organizations must keep data free from corruption, being changed or disclosed [42]. It ensures that data is not modified or tampered with during transmission so that the end-user gets exactly the same information that is available and for which the user is authorized to access.

In a more expanded form integrity can be explained at various levels of existence, operation and sustenance integrity at source, transit, log host, database and analysis [43]. At all these levels, the integrity of the data should remain appropriate. Integrity is ensured at the existence/source/origin level by preventing unauthorized modification of log data at the transit level [44-46].

1.3.3 Healthcare Application Availability

For healthcare IT, consistent availability of healthcare applications is still a priority and is a mandatory necessity. Availability ensures that a system is ready and available for use by an authorized user whenever needed. Healthcare organizations can be seriously affected by shutdowns in applications, hardware or data centres[47].To ensure accountability of treatment and accurate control of patient records, healthcare providers rely more than ever on digital technology. It is important to have vital systems accessible, such as electronic health records, hospital information systems, image archiving and communication systems, and other clinical and administrative systems. It ensures the readiness of correct services to

correct and specific users who are authorized to access the system [48-49].

Availability can be best defined as cumulative measures to help authorized persons have timely access to the system [37]. It is a blend of both authentication and authorization attributes to assure the appropriateness of the user, or a group of users, a computer system etc. To access specified information at the correct time or when they need to perform certain specific operations, except in cases of up-gradation or maintenance of the system. The downtime however should be kept at minimum, but there may be events such as natural disasters or hardware failure when the readiness of the system will be affected.

In most cases a backup site/system runs parallel and in tune with the original/main system so that in cases of downtime of the main system due to certain reasons, the requests could be redirected to the backup system and the entire system should be ready to use. Whenever the main system is not available to its end users, functionality and its effectiveness is not operated properly, causing loss of productive time [50]. On the other hand, more availability will cause the system vulnerable to threats. Thus, effective availability is ensured by arranging the system into hierarchies of subunits so that recovery could be done at a faster rate without jeopardizing the security of the entire system [44].

1.4 Security Risks in Healthcare Web Application

The medical system has been connected to the technological world to ease day-to-day tasks so that the workload of the medical staff could be made simpler. All this is possible with the integration of ICT in the healthcare sector. It is developed to handle patient information to optimize services and contribute to high-quality services for patient care. healthcare web application deals with day-today-services, where

information security plays a critical role. The patient records are perceived as very critical, and this must be handled in such a manner that it is protected and secured from unauthorized access [51]. Although this digital age of technology has its own advantages, apart from these, it has some serious issues. One of the critical issues is “security”, and at an astonishing rate, the case of security breaches is rising. In the field of healthcare, existing research activities are primarily motivated by the goal to ensure the security of all healthcare web applications. The reason behind the rising case of data breaches is flaws and vulnerabilities in the security services available in the application, which leads to exponential growth in data pilferage and data incidents [52-53].

In 2019, it was reported to the government of the United States that the medical records of almost one million people had been breached [54]. In addition, in this pandemic, i.e., covid-19 situation, several problems have arisen in the healthcare web application sector, such as enormous data loads and it has become difficult for security professionals to monitor and secure this data from attackers. One of the main security threats is “compromising on design”, which occurred in several cases [55-56]. Because of the limited period of software development, developers are forced to design software within the defined timeframe that may cause loopholes within the software and create software risks by compromising on design. Security risk occurs as threats may exploit the vulnerabilities of software, causing the failure or damage in software. Evaluating such risk factors in the absence of an appropriate framework may create a challenge [57-58].

The developers still understand and manage risks in haphazard manner rather than implementing a proper mechanism for risk management [14]. For managing these security risks, risk assessment helps to recognize possible threats and their key components. Risk

assessment helps in the identification of risk occurrence with its potential consequences along with the threshold for such occurrences. According to the author of the book 'Risk', who defines risk management. As per his views, risk management is not meant to change the future, not to clarify the past but it requires defining, evaluating, measuring, and prioritizing risks [59]. Meanwhile, healthcare web applications and hospitals are at high risk because they operate with patient personal details, social security numbers, and financial information, along with their name, birth dates, address, and other critical data [60]. These risks may be materialized often by human errors, infringement of privacy policy, software error etc. Knowing the significance, vulnerability, and relevance of hospital data makes it possible to prioritize security steps in this context [61]. In addition, a huge number of internal and external systems are exchanging healthcare data over the network that is also a big security concern. The pictorial representation of data security concerns is shown in figure 1.3.

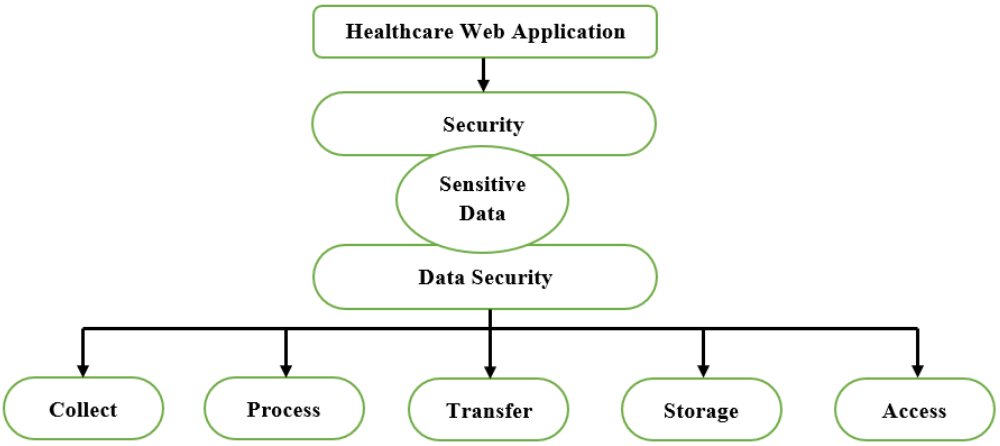

Figure 1.3: Security Mechanism of Hospital Management System

The risks in healthcare web applications create numerous critical security issues. If an attacker gets access to sensitive health records, they can misuse or post them on social media for their benefit or revenge, may

leak the information to the employer of a patient, and can harm by using the patient's identity. Another risk in the healthcare web application is introduced because of outdated software as it may be vulnerable and violates the security rule. Hence, older versions may become an easy target for the attacker. Due to the usage of obsolete software, the United States Office of Personnel Management suffered from security breach cases that revealed more than 4 million current and former federal employees' personal data. In this case, the federal government offered everyone who is affected by identity theft with insurance [62]. These situations may disclose patient's private information and create hazardous situations for hospital institutions where hackers may reveal sensitive and confidential data. This can be done through exploiting vulnerabilities within the system [62], as shown in figure 1.4.

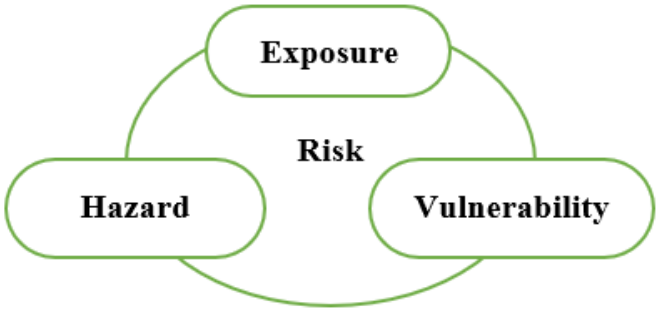

Figure 1.4: Components of the Risk

1.5 Multiple Criteria Decision Analysis

Assessment of security is also a multiple criteria problem. To assess security with its contributing attributes, multi-criteria decision-making techniques will be used in this research work. Multiple Criteria Decision Analysis (MCDA) or Multiple Criteria Decision Making (MCDM) is one of the most important methods for assessment with multi-criteria having multiple levels. MCDA methodology helps in making decisions among the multiple conflicting criteria. In daily life multiple criteria problems can be solved using MCDA methods such as a selection of one criterion from

different criteria. Research based on MCDA methods has a history of only a few years. Further, usage of the internet in everyday life has created a number of problems of multiple criteria. A MCDA problem is generally described using a decision matrix.

1.6 Needs and Importance

As discussed earlier in this chapter, security is not a product or an outcome; it is an entire spectrum of testing activities from the starting phase to the last specifically aimed to identify vulnerabilities within the design and plug the security gaps. In today's scenario where the healthcare web application is immaculate to the lifestyle and industrial needs, an enormous number of applications are launched every day. But, not all healthcare web application systems are tested adequately before release since the software companies have to cater to the excessive demands of users/industries within a short span of time. This increases pressure on the developers, and they frequently leave large portions of these systems untested, which may become a cause for future disasters. Also, there is a huge congregation of healthcare application firms, third party products and open-source applications available in the market, which creates a situation of cut-throat competition among the developer teams. This is also one of the many reasons why it becomes very difficult for testers to run optimum tests on the healthcare web applications before their release.

A survey of about 150 companies conducted by Forrester Consulting in 2010 shows that current practices in the field of application development are not enough to bolster the security of healthcare web applications. The survey also found that a better return of investment is achieved by the industries that follow security practices during SDLC [59]. Security practitioners have proposed various methods of security assessment and incorporation of best security practices during the SDLC,

such as Microsoft Security Development Life Cycle, Comprehensive Lightweight Application Security Process (CLASP) given by Open Web Application Security Project (OWASP), and Architectural Risk Analysis Technical Report by Gary McGraw Software Security Touch-point.

The State of Software Security Report 2017 by Veracode emphasizes on developer's behaviour and security skills gap. The conclusion of the report is that before proper and rigorous testing activities, an application is broadly insecure and is by large prone to malicious attacks, failures, data theft, etc. Statistics of State of Software Security 2017 shows, for instance, that about 77% of software applications have at least one vulnerability on the initial scan, and around 12% are exposed to high or very high vulnerability problems [59]. The report also tells that nearly 88% of java applications have not less than one vulnerability issue.

In tune with recent trends, healthcare web application security is availing attention from experts, research communities and industries. As a solution to the issues with respect to healthcare web application security, security experts should try to discover more approaches to upgrade the security parameters of healthcare web applications constantly. Since the security of healthcare web application is the prime concern, so more efficient and result-oriented practices must be adopted.

If the security is not taken into consideration during the initial phases of software development, the system may be assessed for security only during its implementation. If the project is found to be a failure during the implementation, then the whole task of production has to be repeated, which may result in high cost and rework. As discussed before, the environment in which a project has to work is dynamic and liable to change, so it is clear that security issues also keep on changing. Thus,

security activity is not a one-time exercise but an iterative phenomenon. Users have certain expectations regarding the security of the application they use. These expectations need to be implemented in the form of security mechanisms during the development stage. In addition, a security engineer must think from the attacker's point of view while developing software.

Hence, a comprehensive security framework incorporating all the security attributes is essential to validate across all layers of a healthcare web application without consuming much time to do so. Thus, it becomes imperative for the developers to integrate security features based on a pre-established security policy that embody all security attributes. A well-established security strategy and planning can help to save both time and cost of the development of a healthcare web application.

1.7 Problem Formulation

From the foregoing discussion, it is pertinent that healthcare web application security is becoming a challenging issue in the current era where insecurity is everywhere. There is an extreme need for trustworthy and quality-rich healthcare web applications, which should be capable of preventing many types of modern attacks. A major problem with user's data is that it may be shared with the unauthorised party, and they are not even aware about it. For an adequate secure and quality-rich healthcare web application, there must be an effective and efficient approach that can prevent it from unauthorised access. The main aim is to determine the impact of loopholes and vulnerabilities on the application. The extensive use of healthcare web applications has been weaved into the very fabric of the modern world, where information needs to be secured, and application has to be robust enough to provide both security and reliability. Some of the issues in this regard are listed as follows:

- Unsecure healthcare web applications can lead to information loss, monetary loss, reputation damage, customer dissatisfaction, and even life risk.
- During the designing of healthcare web applications, the question isn't whether we can build a healthcare information management system or not, but the question arises; can we analyse it using security optimization techniques.
- Though different analysis techniques have been proposed, there is still a gap between persisting analysis techniques and newer techniques of analysis for security design.
- Nowadays, many practitioners are identifying new ways to secure healthcare web applications, including decision-making methods, fuzzy AHP and TOPSIS techniques, etc. But they often overlook the security issue of healthcare web applications during the development of healthcare management systems for different organizations.
- Further, securing healthcare web applications is a complex procedure; still, no infallible mechanism exists for qualitative and quantitative assessment. Therefore, there is no proper framework or procedure available for security optimization.

Based on the issues described above, there may be a vast set of research questions that should be addressed. Some of the relevant questions are stated below:

- What are the factors that directly influence the security of healthcare web applications?
- Is there any standard framework available for estimation of security for healthcare web applications?
- What are the major challenges with respect to secure healthcare web applications?
- Can we develop an integrated security analysis plan that incorporates all the security attributes?

- Can we develop a framework that may be used in the design phase to estimate the healthcare web application's security?
- Which security attribute needs to be focused upon according to their respective weightage?
- How general are the lessons learned in this study? Can they be applied in situations involving other environments or organizations with different operational contexts?

Keeping in view the foregoing discussion and issues on healthcare web application, the researcher has formulated the research problem as under:

"Managing Security Risk of Healthcare Web Application: A Design Perspective"

1.8 Research Objectives

In order to achieve the most general goal to develop a secure and trustworthy healthcare web application, the following objectives have been set forth:

- To review and critically examine the literature on existing security estimation frameworks of healthcare web applications and to identify key issues that need to be addressed in the real world.
- To identify security risk factors and security attributes that affect the security of the healthcare web application.
- To establish a relationship between security risk factors and security attributes.
- To examine the impact of security risk factors on healthcare web application.
- To design and develop a novel framework for managing security risk of healthcare web applications.
- To test and validate the proposed framework empirically.

1.9 Methodology Followed

The basic methodology is tantamount to a list of things that we might try in order to reach out the ultimate goal.

- Review of the available literature
- Conceptualization of the approaches
- Development of the proposed approach
- Expert-review and revision of the proposed approaches
- Implementation of the proposed approaches
- Experimentation
- Assessment of effectiveness
- Validation of the proposed approach
- Documentation and finalization

1.10 Significance of the Study

Healthcare professionals and security experts may take help from this research to prioritize the security risk factors in the healthcare web application for creation of secure as well as trustworthy system for the hospital management system. Nowadays, healthcare web application's security has become a major concern for developers. While developing a security mechanism for healthcare web application, the findings of this research will provide security practitioners with ample understanding to adhere to different tactics, rather than relying on informal and traditional approaches. Therefore, the overall impact of this research with its upright benefaction to the field of knowledge may be important, either directly or indirectly in terms of the following:

- It may provide the platform for adequately selecting the effective minimum set of security risk factors.
- It can aid a better understanding of both design and architecture details of the healthcare web application system, which may help to understand the creation and maintenance process.

- It may help to discover the loopholes present in the healthcare web application design at the early stage of the development life cycle through which unnecessary operating cost and security efforts may be slashed.
- It may identify risks in design phase, which helps us zero in on vulnerabilities.
- It may assist in estimating the security of healthcare web application as well as in providing the cost estimate.
- It may help determine the affectivity of the healthcare web application development on the basis of some quantitative evaluation i.e., productivity, lead-time, quality and maintainability, etc.

1.11 Limitations

In order to keep the research precise and within the time boundary, the thesis has few limitations. These are as follows:

- The proposed work has been tested on the small software projects only.
- The proposed framework for analysis can only be applied to the limited number of alternatives and may not be suitable for all kinds of alternatives arising in the future.
- Due to the lack of sufficient data, the proposed model is based on a small set of data.

1.12 Thesis Outline

This thesis is organized in six chapters that cover methods of estimation of the impact of various risk factors, various methodologies used to mitigate problems related to healthcare web application security, apart from annexure, references and other components. In this thesis, researcher has discussed the importance of security risk factors/security attributes and their impact on healthcare web applications. A general overview along with various aspects is discussed in this research. The integrated approach is focused on healthcare web application's security;

in conjunction with a relevant literature survey. Further, the pertinent issue and objective of the proposed research have been discussed.

Chapter- 2:

This chapter discusses the various available approaches, techniques, frameworks, methodologies, standards to produce/build effective and secure healthcare web applications by different researchers around the world. It presents the literature review, basic terminology and several existing security assessments as well as estimation techniques. It also presents the general overview of several security estimation approaches and consequently introduces several issues with healthcare web application security. It has been observed on the basis of review that the quantification of security components are generally untouched or partially included while developing a secure healthcare web application. Therefore, the chapter suggests that security is one of the most important issues while developing secure and reliable healthcare web applications.

Chapter- 3:

This chapter presents the description of security risk factors which is derived from the literature survey. It consists of three phases, i.e., identification of security risk factors. After that, the effectiveness of various security attributes (i.e., confidentiality, integrity, availability, access control and authentication) is illuminated. Further, mapping of these security attributes with security risk factors is described in detail. In addition to this, a discussion about healthcare web application security is presented. The study is carried out to define the various requirements that a security expert considers when choosing an appropriate security engineering approach. It has been observed that for the development of secure healthcare web applications, it is imperative to develop fool-proof security plan that should be focused on security attributes so as to detect more and more faults.

Chapter- 4:

In this chapter, the researcher has proposed a novel security risk assessment framework for the healthcare web application, which melds the security risk factors and attributes of security for the development of secure web application. The framework identifies the security risk factors that affect the security of the healthcare web application. It also expresses how to evaluate the importance and relationship among the security attributes. The ultimate goal of this proposed framework is to secure assets from threats, which can harm the whole healthcare web application. The framework encompasses a complete guideline from identifying security risk factors and their effective security evaluation for designing a secure healthcare web application. In addition, the researcher has also suggested that this approach is suitable for eliciting complete and well-organized security requirements for the healthcare web application.

Chapter- 5:

In this chapter, the concept of validation has been discussed, and methodology for validation has been done in the context of the security estimation process. It integrates all the aspects of security from identification of security risk factors, collection of data with respect to identified factors, evaluation, prioritization and validation of security risk factors. During validation, feedback suggestions have been formulated and have been tested on the basis of statistical analysis. Experimental results are presented graphically. The systematic evaluation process of the proposed framework is conducted systematically. In addition, the empirical validation and statistical analysis of the estimated values of security risk factors have been done by using the security estimation process.

Chapter- 6:

This chapter describes the findings of this research. It presents an overview of the research along with its major findings. In addition, it demonstrates the significant contribution of this research in reference to secure and trustworthy healthcare web application development. It also discusses probable limitations of the research and proposes directions for the future scope of this research work.

Chapter 2
LITERATURE REVIEW

2.1 Introduction

In the fast-growing modern IT world, organizations are heavily dependent on computers for storing sensitive data. With the wide use of computers, day by day healthcare web applications are becoming complex in nature. This is why ensuring healthcare web application security has become one of the very important concerns and a great challenge for security practitioners [63]. During the last few years, a lot of data theft has been reported, and now healthcare web application security is becoming a predominant concern. Moreover, security flaws in design may cause the application to violate its security and results in unauthorized disclosure, modification, and data destruction. Hence, in the current scenario, ensuring security at the early stages of the development life cycle has gained much attention.

In the software development life cycle, the design phase prepares the structure of the software, in which software is at a very young state. Hence, in this phase, controlling vulnerabilities and improving security has become one of the major considerations [64]. Discovering vulnerabilities early in software may reduce cost, rework, and time for later phases of the software development life cycle. The following section discusses the overall work done to fulfil the project's objectives in terms of findings and major contributions to strengthen the need to apply a measure in the early development life cycle.

It is observed from the literature survey that significant efforts are being made for the development of secure healthcare web applications. It is generally related to building authentic, durable, and less vulnerable

applications, where security is considered as one of the major factors. Two main groups of researchers have made significant endeavors for developing secure healthcare web applications. The first group believes in spraying security in the healthcare web applications after its development. This process is costly and ineffective in removing the security loopholes from it. In contrast, the second one believes in incorporating security features throughout the development life cycle. This reduces the rework and is cost-effective. A survey has also reported that the majority of security metrics access security at the system level. Researchers also believe that some existing metrics can help in depicting the immunity and reliance of the system, but there is a need for more research [65].

2.2 Expert's Saying

According to McCurley et al., measurement can be enforced into the development life cycle to track and enhance the security properties of software under development. The explanation and investigation of relevant measures help to diagnose problems and identify solutions [66]. According to G. McGraw, for building secure software, security must be addressed from the early stages of the software development lifecycle instead of spraying security features on the security holes. As "no one can test the quality into a piece of software," so there is a need to address security issues from the ground up and educating security practitioners about how to build secure software [3]. As C. Kaner and W. Bond stated, there is a need to answer various questions to decide which software security metrics should be used in a particular situation. They also noted that measurement is the procedure of quantifying the software attributes to describe them through a properly explained approach [67].

D. Taylor and G. McGraw suggested that the measurement and metrics should be incorporated from the early phases of development life cycle [68]. In support of the above, S. Chandra and R. A. Khan stated that

the design phase is the most suitable phase for the estimation of security. The estimation of security at this phase will help to defend software from loss [69]. Lord Kelvin wrote that “if you cannot measure it, you cannot improve it” [70]. S. Jain and M. Ingle suggested that for increasing the security feature of a software product, metrics may also act as a checklist [71]. According to C. Wysopal, security defects are part of software development [72]. Schult et al. in 1990 stated that vulnerability is a weakness through which a hacker may bypass the security measures [73]. B. Patrik and J. Per stated that the GQM procedure is one of the most accepted goal-oriented procedures, which is highly recommended for the development of metrics framework [74]. According to A. Agrawal and R. A. Khan incorporating security decisions in SDLC, the design phase is most appropriate phase [75].

2.3 Existing Mechanism

This section briefly represents some of the relevant contributions of security practitioners and researchers.

Abushark et al. (2021) defined several taxonomies and created a design hierarchy. It incorporates the most prevalent quality evaluation factors, which integrate variables, characteristics, and traits from different Security Requirements Engineering (SRE) methodologies. The fuzzy AHP-TOPSIS model is utilized in this paper as an MCDM (Multiple-Criteria Decision-Making) model. The usability of SRE has a huge impact on the security requirements consistency. The author defined STORE technique as highly consistent and usable approaches among all other SRE techniques with a threat-driven approach. In addition, they have concluded that STORE elicits security requirements in an efficient and well-organized manner [76].

Rajeev Kumar et al. (2021) identified and analyzed the characteristics of security and sustainability. In this study, the fuzzy AHP

algorithm is utilized for quantitative assessment, which is verified by four other approaches based on AHP. As a result, the evaluation of security in this study will assist developers in formulating standards that will ensure the development of more secure online applications [77].

Abdulaziz Attaallah et al. (2020) have discussed security as a critical aspect in the process of software development that must be considered during its development cycle. Thus, the researcher evaluates the effect of security risks using the integrated approach of TOPSIS and fuzzy AHP. This hybrid approach is ideal for evaluating malware analysis on the basis of its impact. In the current study, the author collected different IT specialists who belong to diverse educational backgrounds and software industries. This empirical investigation uses the hybrid technique and evaluated numerous security risk factors at the design stage on 10 institutional web applications. According to the evaluation report, among the 10 institutions, the 8th institutional web application was determined as the most efficient and durable security system among all competing options. Finally, researchers suggested that security is a key aspect that must be taken into account from the initial phase of software development [78].

Fahad Ahmed Al-Zahrani (2020) has reviewed the healthcare application to ensure software usability and security by using the hybrid technique. The author has suggested that security experts must design a healthcare web application with two intents; it ensures usability, given to fulfil the users rather than ensuring optimum security and efficacy of security as well as usability is in the early development phase. In this paper, the author has defined 17 criteria, 4 in level 1 and others are in level 2 with 6 alternatives to evaluate HMS software. In the proposed study, fuzzy logic, ANP (Analytic Network Process), and TOPSIS technique were used to promote healthcare applications with maximum

protection while maintaining its usability derived from empirical review of comprehensive data. The author has suggested that this technique is used for the assessment of healthcare web applications. In addition, this integrated technique evaluated that the 5th alternative of HMSSS (Hospital Management System Software Services) provides optimum security among all six alternatives with the highest user satisfaction [79].

ZhihanLv and Liang Qiao (2020) have performed research on the basis of healthcare big data for predicting risk in the healthcare sector of China. Data security and privacy issues have risen with the exponential growth of healthcare records. These security issues hinder the full transition of healthcare data tracking. To predict risk and its security measures, two security factors were used that are evaluated by using a questionnaire. The questionnaire key questions are categorized into various aspects, and the outcome is used to measure the threat level by using the Likert scale 5. Furthermore, the Delphi technique is used for the assessment of privacy and security risk to build an indicator framework. This study suggested two levels of privacy security mechanisms: technology and management. It suggests that healthcare institutions have to pay attention to the security of data privacy and digital medical data usage [80].

Saleh M. Altowaijri (2020) proposed a framework for the healthcare sector to enhance the healthcare security of cloud computing. Now, security provision for cloud data is also becomes a tedious job. This technique is widely adopted to access EHRs (Electronic Health Records) from any location at any time, create and manage these sensitive data that are in encrypted form in the cloud. Such information is susceptible and a lucrative goal for fraudsters. To secure these data, several mechanisms have been investigated to secure cloud data, including Authentication, CIA triad (Confidentiality, Integrity, Availability), and Non- repudiation.

To solve this dilemma, the author has obtained assistance from Big Data, and he introduced the concept of master nodes and slave nodes in his architecture to store the data. In this architecture, the master node keeps metadata, and on the other side, the responsibility of the slave node is to store data. The sensors can access all consumers' data and ensure its efficiency as it is in the form of quasi-structured, and these data are easily accessible. This architecture stores data in encrypted form. It is based on the RSA (Rivest Shamir Adleman) algorithm and PKI (Public-Key Infrastructure) algorithm that provides accessibility to authorized users at a certain time to access data of particular patients [81].

Jasleen Kaur et al. (2020) have identified a few security risks at the design phase and its risk assessment in healthcare web applications. Using artificial neural network technique based on inference system of Adaptive Neuro-Fuzzy Interface System, i.e., ANFIS, risk assessment is conducted. This paper suggests that “security” risk is a major concern for researchers, which might jeopardize healthcare's web application security. Globally, most of the patients were impacted by security breaches in the healthcare system. For ensuring security in healthcare web applications, researchers have used the ANFIS technique. The eight major security risks are addressed in this paper during the early phase of development in healthcare web applications. These security risk factors are evaluated by using ANFIS neural technique and validated by fuzzy regression modelling. Finally, this study concludes the quantification analysis of security risk in order to make it easier for developers to handle security threats occurring at the phase of design. In addition, they have suggested that this methodology will be helpful because it is designed on the basis of experienced researcher’s and security expert’s perspectives [82].

Hathaliya and Tanwar (2020) reviewed the privacy and security factors of Healthcare 4.0, where 4.0 makes use of Fog Computing (FC), tele-healthcare technologies, Cloud Computing (CC), and the Internet of Things (IoT) to share data among different stakeholders. Despite these advances, maintaining security has always been a difficult issue and could result in a data breach through which hackers acquire complete access. To prevent these security issues, the author defined the security taxonomies in a structured manner. This taxonomy covers various privacy and security aspects including IoT, Machine Learning (ML), tele-healthcare, network traffic, authentication scheme, and Wearable Devices (WDs). The author also investigated the blockchain-based approach to provide vision to both practitioners and scholars. In the future, the authors have suggested the blockchain-based secured decentralized architecture will be explored for different smart applications [83].

D. Praveena and P. Rangarajan (2020) have proposed a new framework for machine learning, which is designed to provide hybrid cloud networks with security when data is stored, accessed, or retrieved from cloud databases. This study comprises of two sub-sections namely: mechanism of access control and encrypted deduplication. To prevent duplication, two new algorithms are proposed called the DSRBACA based on Dynamic Spatial Role and Access Control Algorithm and Deduplication Processing Algorithm (DDPA) to limit access to user data integrated with an existing Enhanced C4.5 algorithm. This is a newly proposed algorithm; both are used for safe storage and retrieving data to eliminate redundancy or any duplication in the cloud database with limiting access. The outcome of this proposed architecture is that DSRBACA performs well in limiting the cloud user's number as compared to the traditional RBAC model and provided 90% accuracy in prevention and detection. The key benefit of this new application is that it reduces

the security risks of a hybrid cloud and ensures the security of data with limiting time and space-based access to data [84].

Israa Abu-elezz et al. (2020) have investigated healthcare block-chain technology's scoping review with the strengths and risks. It is an advanced structure of data system consisting of huge list records in blocks. This research is carried out in three phases: identification phase, screening phase, and eligibility phase, and these filtering stages were done through a flow diagram of Preferred Reporting Items for Systematic Reviews and Meta-Analysis (PRISMA). In this study, 84 studies have been used for the initial search and extracted the result from the five databases, where 70 articles were selected that are unique and the rest duplicate were eliminated. After the completion of the second phase, 64 articles are considered in screening, and 6 publications were excluded from newspapers and magazines. Further screening in the final phase resulted in the rejection of 30 articles. The reason is to focus on the methodology and design of blockchain implementation in healthcare, and the integration of different technologies with blockchain did not meet the criteria of inclusion. This study sums up with 37 studies with eight advantages and threats. The advantages were divided into organizational or patient-related benefits. Researchers have suggested that this analysis will help to obtain a more precise understanding, owing to various constraints. The findings of this analysis must be viewed with caution, and this scoping review provides useful insights, particularly in medical care [85].

Yang Lu and Richard O. Sinnott (2020) have addressed the need of privacy and security solution for the management of online data in a smart healthcare system. A drastic change from conventional to Electronic Health Records (EHR) in database systems is taking place in healthcare institutions and becoming increasingly important. This enhancement in the

medical field introduced several security challenges that can threaten personal privacy and device security. The aim of this research is to balance privacy and its utility in the centralized system while seeking to exchange, visualize health data, and integrate with healthcare departments. This paper demonstrates a systematic review in s-health related to privacy and security solutions such as authentication, privacy-awareness, access control, and anonymization. The key features are described in s-health, where CIA principles reflect the general criteria for security and privacy. The researcher picks mobility, richness, complexity, and novelty to evaluate related solutions in terms of compliance with s-health services. This paper explores privacy-preserving and security solutions for meeting privacy criteria built in the contexts of s-health as well as maintaining service quality in a data-rich environment. In addition, the authors have mentioned that future studies are still necessary for the development of current solutions to provide streamlined, real-time services. The analytical technique must be lightweight; a cryptographic algorithm must be highly efficient, which maintains integrity and confidentiality [86].

Zhihui Wang et al. (2020) have discussed the fabrication process of Wearable Health Monitoring Systems (WHMS) and the CEPAK method to protect the data of wearable devices. These devices have advanced micro-sensors that are integrated with IoT technology to compute and store user data. Despite of these computing services, WDs pose serious security risks, including the DoS attack (Denial-of-Services), replay attack, server-spoofing attaches, and modification attack. Thus, the WHMS comply with the following basic security characteristics: anonymity, key control, mutual authentication, forward secrecy, low delay, light-weighted, etc. Key agreement and dynamic identity authentication protocol are proposed in WDs named CEPAK protocol to fulfil these characteristics. In order to deal with attacks on network and computing

resource consumption and communication resources, the CEPAK protocol's capabilities are evaluated. It can be seen that the authentication server of cloud-assisted can significantly share and save the consumption of computing resources efficiently. It also increases the system's ability to handle triggered attacks by computing resource consumption on the principle of achieving the system's security requirements [87].

K. Vijayakumar and V. Bhuvaneswari (2020) have addressed the skewed and unreliable obstacles in the applications of healthcare. The IoT (Internet of Things) provides a new path for the healthcare sector, i.e., the use of actuators and sensors that are feasible to send information without manual assistance and boosted the traditional model of medical. IoT technology offers tremendous advantages for diagnosis in healthcare. This paper has also mentioned the IoT application challenges in the healthcare sector, such as connectivity, security, and hardware-related issues. It's time to build the new framework, which handles the various technology exponential growth and its risks. To overcome and solve a real-time business dilemma, the three layers are proposed in the new framework, i.e., the Sensor layer, the Network Layer, and the Service access layer. In the communication layer, the most significant threats are sent to the network layer by sensor data through wireless communication, whereas in the third layer, unauthorized access and malicious are the most severe risks. In addition, they have mentioned that these risks can be mitigated by properly implementing authentication, audit, and authorization [88].

Wajdi Alhakami et al. (2020) have discussed the model for managing the data of healthcare with the use of a symmetrical framework in the view of cybernetics. The healthcare sector needs to explain its infrastructure and digital transactions from a new viewpoint. Several researchers examine the latest techniques based on the portability of the

Health Insurance Portability and Accountability Act named HIPAA. The HIPAA approaches are completely updated and stable for the modern and smart healthcare system, but the census report of cyber-attack is at an alarming state. To provide a secure platform for the healthcare system, the authors depict a symmetrical model that will allow experts to protect it against attack with the use of the symmetrical model. This model is based on the author's name and is known as an ARAR model having five steps; initiation, classification, assessment, result, and application. The aim of this model is to create a productive digital infrastructure for healthcare from a new viewpoint. The authors have mentioned this model would be effective in achieving the objective of handling and comparing its various sources and its potential approximate variety to minimize the problems of variety [89].

Tang Yongjun (2020) has proposed a new KNN-BP algorithm for IoT (Internet of things) application and security design by integrating a neural network of BP and KNN for COVID-19. The risk on the network system will drastically increase when communication ports are limited and can be partially fulfilled and managed by existing authentication methods. It can be done by using a prediction/classification algorithm including both the KNN and BP algorithm; KNN stands as the K-Nearest Neighbours' algorithm. It is based on the non-parametric method, whereas BP stands as the Back Propagation algorithm of the neural network. The model of the neural network has reverse error feedback characteristics, including a multi-layer back-propagation function. The finding of the experiment showed that the BP neural network has a high capacity for prediction as well as have a certain capacity for fault tolerance. Although, in the case of abnormal test sample attributes, the back-propagation algorithm demonstrates significant inadaptability and reduces the prediction accuracy, whereas the KNN-BP algorithm provides high stability. Eventually, the authors have concluded that the sensor gathers a

huge amount of data in the environment of IoT applications, which may provide a huge sample number for the prediction [90].

Wortman and Chandy (2020) have discussed on risk-based tool, i.e., SMART tool for evaluating the security of system designs. This tool basically works in three phases: On the basis of the design model of the original system, the user can generate either one or multiple attack graphs. This tool can then read a given device security model for attack graph to each user. In the second phase, the SMART tool starts gathering the requisite data to accommodate the variables of adversarial risk and for defensive cost, including the set of other non-CVE (Common Vulnerability Exposures) information, CVE information set for API (Application Programming Interface) and element-specific variables. In the last phase, on the basis of the information set, the SMART tool performs the SR (Security Risk) calculation for the given model [91].

Alka Agrawal et al. (2020) have analyzed software security to ensure software usability and security by using the hybrid technique based on Design Tactics, i.e., F-ANP and fuzzy symmetrical TOPSIS. In this research, three main attributes i.e., resist attacks, detect attacks, and react and recover from attacks, have 15 sub-components; each attribute has 5 sub-components. These attributes are selected at two different levels, conducted with 10 different institutes, where F-ANP evaluates the criteria weight, and TOPSIS defined its impact. It is concluded that the first institute among all 10 competitive alternatives offers the most effective and robust safety mechanism. In this report, the proposed symmetrical evaluation would allow both developers and designers to prioritize security factors and categorized the attributes of security, including security tactics importance during the life cycle of software [92].

Jinfeng Li (2020) has discussed the recent developments in SAST (Static Application Security Testing) and introduces a new vulnerability mapping matrix based on synchronizing the top 10 vulnerabilities and 25 software risks. These top vulnerabilities and software error lists are provided by industry-standard OWASP (Open Web Application Security Project) and SANS/CWE (Common Weakness Enumeration) synchronized in a matrix with queries of Checkmarx vulnerability. This provides a security framework for applications that enables development teams to minimize false positives, review to fix code vulnerabilities found in penetration and static scans and target the results for greater accuracy. On the basis of SAST, Checkmarx helps in decision-making in the remediation of bugs and mitigation of vulnerabilities with enhanced performance [93].

In 2020, Mohammad Zarour et al. have mentioned lack of maturity in software security leads to a growing number of cyber-attacks. At present, their success rate is very likely because of the low quality of software. They also mentioned that different practitioners and researchers combine certain activities in secure software development, which makes it vague and create flaws in it. Researchers have suggested two phases to resolve this problem and to address these mixed-up activities i.e., Requirement Engineering Phase and Design Phase. In the phase of requirement, security goals must be identified and translated into NFR (Non-Functional Requirement) to conduct risk analysis using the CORAS method. The Design phase builds a secure architecture considering the Functional User Account and Security Design Analysis. In addition, researchers have mentioned that more research is still needed to enhance the software's analysis and design process to reduce the final software's vulnerability [94].

Kewei Sha et al. (2019) have conducted a detailed survey on current edge layer IoT (Internet of Things) security solutions and to inspire more IoT security designs based on edge computing. In several IoT applications, the reported security breach cases indicate that the application of IoT could threaten physical systems. There are two main causes in IoT applications: inappropriate design of security and extreme resource constraints that creates several security problems. Though there is a need for robust security design, and it is difficult to secure IoT. This research is conducted in three folds: the general architecture of edge-centric IoT is in the first fold, in the second fold contains a detailed survey, and the last contains identified challenges and future work [95].

Liezel Cilliers (2019) has investigated the issues of privacy and data security by using wearable healthcare devices. As wearable devices provide the earlier identification ability to track the status and detect adverse patterns that facilitate chronic disease detection. To make it useful for individuals, wearable devices gather information in two forms, either automatically by using sensors or manually by the user. Usually, the organization retains data collected by a wearable system in a centralized database, which will reveal all user's information if there would be privacy violations. It is based on three fundamental information security pillars: CIA (Confidentiality, Integrity, and Availability) triad. This research uses a quantitative approach for the survey, with a convenience sampling method for recruiting survey respondents, and on the other hand SPSS V25 is used to analyze data using descriptive statistics [96].

In 2019, Mamdouh Alenezi et al. have explained their views about software security risks, where security is a major concern in software development. The use of best practices during the SDLC is necessary. The evaluation of risks should be carried out from the starting phase of the

development phase and must be carried out as a continuous process that is incorporated into each cycle phase. It should start from the Pre-requirements phase to Release and Maintenance Phase. This paper illustrated the types and the classification of risks that have to be evaluated and mitigated at an early stage to establish a SSDLC (Secure Software Development Life Cycle). However, they also provide a qualitative report on the practical realities of software security risk that helped software developers to prepare and enhance their performance and strategies [97].

Zhenzhen Nong and Sally Gainsbury (2019) have addressed social cues present in the online environment, which makes difficult for an individual to check their safety and recognize latent threats in online activities. This research will direct future studies in evaluating the effect of a website on online risk-taking behavior by identifying environmental indicators [98]. In the same year, Alisdair et al., have published an article about how online scam is a big threat to cybercrime. The authors explain that the fraud occurred well before the internet came into existence, and the most prevalent types of cybercrime radicalized on the internet. They illustrate it can be difficult to protect the offender's custody without a specific description of online fraud. The directive must ensure the same basic framework for all countries within the Union. Therefore, the directive must provide for a minimum standard of security across the Union. They must ensure a minimum standard of security for EU people for effectively tackling online fraud. There is a need to correlate the laws and set out a framework to protect all online fraud [99].

Jasleen Kaur (2018) addresses the several critical security risks that might penetrate the software's design phase. These security risks are collected from the CWE (Common Weakness Enumeration) list; is a community that eases the development procedure of software along with

security. CWE provides a set of hardware and software vulnerabilities by offering standard security tools for detecting, preventing, and mitigating the different weaknesses of software and hardware. The ten security risk factors of software are enlisted in this paper that is filtered from the CWE list; these factors are based on CIA (Confidentiality, Integrity, and Availability) along with Access Control. The researchers suggested that if these security issues and loopholes are addressed and handled at the early phase during the development, then it would significantly help to mitigate security breaches and can lead to more secure, efficient, and reliable applications. It will reduce the incidence of such attacks through early detection and mitigation [100].

In 2018, Hala Assal et al. investigated SDLC procedures to explore real-life practices of software security used by developers. Researchers have noticed on different approaches to software security and best practices are frequently disregarded. The discussed practices of software development are affected by several reasons such as lack of security knowledge, company culture, labor division, and availability of resources. It affects the team's strategies, which affects the team burden as well as business culture, external pressure, security awareness, and other influential factors. Instead of blaming developers, the authors illustrate the need for a new lightweight practice that considers the development realities and pressure without compromising the security [101].

In 2018, Raman Shapaval et al. have provided a detailed reference model for the management of security risk in IoT systems. This reference model is based on a domain model, i.e., ISSRM (Information Systems Security Risk Management) model. The ISSRM model defines three major conceptual pillars namely secure assets, their security risks, and countermeasures. The researchers examine several vulnerabilities and their potential countermeasures in the Open Web Application Security

Project (OWASP) that will assist in the detection and management of IoT system's security risks. As IoT systems rely heavily on internet computing and the cloud, it required more work to highlight threat agents and their effects. Although researchers reinforce this model in future studies with the concept of explicit IoT assets, identification of risk countermeasure including the analysis of security trade-off [102].

In 2017, Shams Tabrez Siddiqui has discussed the different security metrics that are important during the software development phase. Security factors are generally considered as a post-development task by software developers. When the security measures are evaluated from the beginning, mostly during the design phase of the SDLC, the security threats are measured more effectively. These security metrics define numerous security activities like security training, build secured use cases, describing security models, security testing, and reviewing it, and more other factors that help to build secured software. The researcher suggested that the proposed metrics of security must be integrated into the initial stage of SDLC, and even these metrics evaluated its security requirements ratio, released patches to fix security vulnerabilities, omitted the exceptions ratio, etc. [103].

2.4 Findings from the Literature Review

During the literature survey, we found that despite the necessity of establishing security during development, especially at the design phase, there has been a gap between the two attributes of security. To fill the gap, the relationship between security attributes and risk factors should be established. The available literature can be divided into three patterns: in 1st category, the approaches that strive to improve security during the SDLC. The approach of second category tries to enhance the application's security either after the development or at later phases of development. The approaches in the third category improve security through MCDM

techniques. The limitations of the first and second categories are being too late to manage security during or late stage of the development life cycle. The identified approaches in the third category are important and can be used to estimate security to enhance the application's working life. Unfortunately, no work is identified to estimate or predict security at an early stage of the development life cycle. Hence, there is an urgent need to develop a security-centric approach to improve security for the duration.

After a careful and centric study of some available approaches, the following inferences are drawn:

- In the field of healthcare web applications development, every organization follows its own mechanism for securing healthcare web applications. There is no common mechanism and framework exists.
- Users want secure healthcare web applications so that it is not interrupted when the security shuts down.
- There is a need to collect expert advice in this field and develop a framework that provides security to healthcare web applications.
- From the literature review, it is clear that MCDM approaches are the most important techniques that are useful to assess the collective qualitative data quantitatively.
- There is no single work ready for use for the prediction of healthcare web applications, so there is a need to estimate security at the early stage of the development process to minimize the maintenance.
- Design phase is the essential phase of the software; this phase is most responsible for security. Hence, assessment of security in the design phase is important.
- To bridge the gap between security and its attributes for assessment and quantification of security is needed.
- For measuring or improving security at the design phase, current measures depend greatly on threat models and attack types. They offer

little information when the environment changes. While MCDM approaches can give better results.

- Measurement of security for the healthcare web applications is hard. That is why; it is rarely seen, especially to measure security for improving the life span of healthcare web applications.

The process of integrating security in the software during its development is called secure software development life cycle. With the increasing demand for secure healthcare applications, security professionals are facing new challenges to fulfil requirements of customers while developing software. From the security perspective, healthcare web application development includes security strategy, security design, security attributes, and security management. Unfortunately, security is often integrated only in isolation and late in the process. Corporations impose development constraints due to cost, time-to-market requirements, customer satisfaction concerns, productivity impact, etc. This gives result in the improper development of secure applications with less durable security. Security is the most demanding concern in the present era and this demand is making companies successful or unsuccessful in the market.

2.5 Conclusion

This chapter reviews several security frameworks for healthcare web applications proposed by different authors. Several key concepts of different security engineering approaches have been discussed. The researcher also discussed several security challenges that are raised by various security experts. One of the most critical and important healthcare web application development activities is security; poor execution can lead to complete failure of the overall system. The efficiency of healthcare web applications depends on security requirements and how much the product meets the requirements. Therefore, it can be said that

incorporating security in healthcare web applications requires an extensive skill set combined with experience to be performed well. A framework is needed that not only identifies the core problem area but also suggests the guidelines and the solution to these core problem areas. To achieve these goals, an efficient security framework for healthcare web applications is proposed in this research study that gives a suggestive solution to these problems.

Chapter 3

IDENTIFICATION OF SECURITY RISK FACTORS

3.1 Introduction

The increasing number of Internet users has created an environment where software plays a crucial role in all kinds of information exchange. This leads to increased demand for different types of software and their respective uses in conducting all types of information interaction. The role of software is very important for any application/organization; hence its security cannot be undervalued in any way. Security is an activity that exposes whether the security functions are correctly implemented and whether software behaves correctly in the presence of a malicious attack [104, 105]. Making software secure is not only related to the reliability and confidentiality of a system that contains important or personal data, but it is also essential for giving better results. Therefore, the security of software is vital before it is being implemented in real-life systems. Security is performed at various phases of the SDLC, starting from requirement and analysis through design, implementation and verification [106, 107]. Finally, it points out the future focus and development directions of the security framework. In this chapter, we discuss various security risk factors as well as security attributes related to the healthcare web application.

Software security deficiencies do not come to the surface as easily as other faults and errors found during testing [108]. Therefore, software security is required to identify defects and faults that are rather difficult to make out. The security framework is performed to make sure that the software under test is satisfactory robust and works in an adequate mode even at the time of a malicious attack [105, 109]. Compound systems are hard to test, and therefore the probabilities of getting untested portions

are found. These untested portions act as loopholes through which a breach could be made in the software, which may affect the efficiency and capability of software and may also result in the loss of important information. To overcome such types of problems extended security framework needs to be developed, which may help to identify and address different types of security breaches in the design phase of the software development life cycle.

The main aim of security is to make sure that sufficient attention is given to the healthcare web application to identify the security risks and perform reasonable tests to ensure the proper functioning of the applied security measures. It also ensures that plenty of expertise exists to carry out adequate healthcare web application security. It is one of the activities that are used to reduce vulnerabilities within a healthcare web application system and control potential future costs. This signifies the security of healthcare web application adherence to its function as well as non-functional constraints.

3.2 Security Risk Factors

The demand for security, like security and other software attributes is growing day by day. As the customer's priority has drifted towards security along with other quality attributes, the developers are also focusing towards the same and striving to develop secure software. As information about an organization's assets is processed through software, security concerns for software grow more for the organizations. To estimate security and to improve it, organizations need to identify and address the different types of security characteristics that affect security directly or indirectly. Software security may be enhanced by identifying and mitigating security risks at earlier phases of SDLC.

Security is typically thought as a combination of two elements, i.e., effective risk management as well effective countermeasures [110]. One of the key causes of security breaches is security risk, which must be addressed appropriately in order to produce secure application, and for the development of secure application, this should be addressed properly [111]. On the basis of literature review and expert's opinion some of the most common security risks are shown in table 3.1.

Table 3.1: An Overview of Security Risk Factors at Design Phase

S. No.	CWE ID	Security Risk Factors	Scope Factor
1.	CWE-767	Access to Critical Private Variable via Public Method (ACPVPM)	Integrity, Access Control
2.	CWE-260	Password in Configuration (PCF)	Access Control, Authentication
3.	CWE-311	Missing Encryption of Sensitive Data (MESD)	Confidentiality, Integrity
4.	CWE-620	Unverified Password Change (UPC)	Access Control, Authentication)
5.	CWE-366	Race Condition within a Thread (RCT)	Integrity
6.	CWE-426	Untrusted Search Path (USP)	Confidentiality, Integrity, Availability, Access Control
7.	CWE-494	Download of Code without Integrity Check (DCIC)	Confidentiality, Integrity, Availability
8.	CWE-362	Concurrent execution using shared resource with improper synchronization ('Race Condition') (RC)	Integrity
9.	CWE-454	External Initialization of Trusted Variables or data stores (EITV)	Integrity
10	CWE-915	Improperly Controlled Modification of Dynamically-Determined Object Attributes (ICMD)	Integrity

3.2.1 CWE 767 (ACPVPM)

This weakness might allow an attacker to modify the variable that contains an unintended value. It is a public method that reads, alters, or modifies a private variable; it may violate other code part's definitions or values. In addition, if an attacker can read the private variable, it is easy for the attacker to launch more attacks as well as can expose sensitive information, and it affects the integrity scope of variables [112]. This vulnerability may allow an attacker to access a private variable, and it

would also affect the code integrity if a variable can be modified by public methods [113].

Example:

```
private: float price_amt;
public: int updatePrice(float newRate)  {
price_amt= newRate;
}
```

In the above code, the variable price_amt is private, but for any updation, the variable is accessed within the public function.

3.2.2 CWE 260 (PCF)

This weakness allows the software to keep passwords that could be accessed by actors from software configuration, even if actors did not know it. Even, it creates worse conditions on system security, i.e., attackers retrieve the password, obtain access to potentially sensitive data, and reset passwords [114].

Example:

```
<connectionStrings>
<add name="SQlServerConnect" connectionString="Data Source=local host; Initial Catalog=UserDatabase; UserID =UserId; Password= UserPwd;" providerName= "System.Data.SqlClient"/>
</connectionStrings>
```

As we have seen, the configuration file contains the login and password for a database and stores this data in plaintext without any encryption. This flaw is also present in the Java Web Configuration file where the password is kept in cleartext format.

This vulnerability is introduced in “Medfusion 4000 Wireless Syringe Infusion Pump”, but this software is not allowed for external

communication. If the configuration file is allowed, then it can be successfully exploited and remote attackers easily accessed data in an unauthorized manner. The impact of this flaw is low [115].

3.2.3 CWE 311 (MESD)

This vulnerability introduces due to the absence of proper data encryption, which allows transmission of the assurance of confidentiality, transparency, and integrity by properly enforced encryption. Before storage or transmission of data, the application does not encrypt critical and confidential information. This vulnerability is triggered during the phase of architecture and design due to lacking of security tactics.

In order to communicate or access sensitive information, the following code is intended to link to a site. The below URL connection is not encrypted, and unintentional actors will read all confidential data which is sent or received from the server [116].

Example:

```
try {
String urlString = “http://www.xyz.com”;
URL url = new URL(urlString);
HttpURLConnectionUrlCon = (HttpURLConnection) url.openConnection();
UrlCon.setDoOutput(true);
UrlCon.setRequestMethod("PUT");
UrlCon.connect();
OutputStream out= new OutputStream();
UrlCon.getOutputStream();
UrlCon.disconnect();
}
Catch (IOException e) {
throw new RuntimeException(e);
}
```

Recently, Apple XCode Software is affected by this vulnerability. It exists in the debug session due to missing communication encryption during the network [117].

3.2.4 CWE 620 (UPC)

An attacker might use this flaw to alter victim's password and allow him to gain access to get the user's rights. On the other hand, application does not require any kind of authentication or knowledge about user original password when user creating a new password [118].

Example: A user's password is changed using this code.

```
$UserID = $_GET['userid'];
$UserPass = $_GET['userpass'];
$checkpwd = $_GET['checkpwd'];
if ($'userpass'== $'checkpwd') {
SetUserPassword($UserID, $ UserPass);
}
```

This code ensures that the user inputs two times the same new password but does not validate the authorized user request to change the password [40]. This flaw exists because the online application does not verify an old password and it might be possible for an attacker that he can obtain as well as control the user account by requesting to change the password. This vulnerability existed in Version-4.1.402.34662 of LenovoEMC NAS Firmware Software and its impact is low [119].

3.2.5 CWE 366 (RCT)

In a multi-threaded environment that uses the locking functionality around code that enforces to block, alter, and read persistent data. If a resource is used concurrently by two execution threads, there is a risk that invalid resources can be used.

The code mentioned below creates a race condition that is introduced in programming when the critical resource is changed by two or more execution threads [120].

Example:

```
public class RacePrgm{
static int foo = 0;
public static void main(String[] args) {
RaceThreadth=new RaceThread();
Thread firstThread=new Thread(th, "First thread");
Thread secondThread=new Thread(th, "Second thread");
firstThread.start();
foo=1;
secondThread.start();
}
public static class RaceThread extends Thread {
public void run() {
System.out.println(foo);
foo=foo+1;
}
}
}
```

3.2.6 CWE 426 (USP)

This weakness might allow attackers to access data files in an unauthorized way or unexpectedly change configuration settings to execute their programs. Such programs will enable the application to find critical resources using a search path, then that search path may be altered by an attacker through malicious code.

The given code will take a user's name by using 'ls-l/home' command to get the content list and directory of that user's in-home

directory. This code is also susceptible to a PATH attack because through ps or grep commands, an attacker can generate malicious commands. Although there is no explicit privilege in the program to execute the commands of the system, the interpreter of PHP can run with higher privileges than users by default [121].

Example:

```
//assume the function getCurrentUser() returns a name of user
$UName = $_POST["user"];
$cmd1 = 'ls -l /home/'.$UName;
system($cmd1);
$UName = getCurrentUser();
$cmd2 = 'ps aux | grep ' . $UName;
system($cmd2);
```

In December 2020, Foxit Phantom PDF and Foxit Reader for Windows were remotely accessible through this vulnerability. The vulnerability occurs in the directory of current working when the installer searches for taskkill.exe file. At that time, a remote attacker will trick the victims into running a remote SMB (Server Message Block) share installer file and executing arbitrary machine code. SMB share provides mutual access between nodes within the network and allows computers to read and write files via LAN to a remote host. This vulnerability's impact is medium in Foxit Reader for Windows and PhantomPDF [122].

3.2.7 CWE 494 (DCIC)

The main drawback of this weakness is that the program's executable code or source code is downloaded without verifying its data integrity and origin. The absence of authentication makes it possible for attackers to fool the machine by executing malicious code or altering the source code. It provides different possibilities for attackers to execute attacker's instructions, interpret or modify potentially sensitive

information, and render software worse for legitimate users. This vulnerability is introduced during the phase of Architecture, Design and Implementation phases [123].

Example:

```
public static void main(String[] args) throws Exception {
   // get the URL of jar that contains the target class
URL[] urls= new URL[]{new URL("name of new URL")
};
// New URLClassLoader is created
URLClassLoaderloadurl = new URLClassLoader(urls);
//Target class is Loaded
Class targetClass = Class.forName("loadMe", true, loader);
}
```

This java code doesn't validate that the loaded class is the intended class; attackers may alter it to execute malicious code [123]. This weakness has recently emerged in routers NETGEAR R6700; an attacker is able to compromise the affected device. Due to the lack of software integrity, vulnerability occurs if updates are downloaded. An attacker can remotely gain access to the local network and execute a man-in-the-Middle attack to control the affected device entirely after a successful update of the software. Its impact is not too much high, but at present, its patch is not available. The security expert has no official solution to this vulnerability at present [124].

3.2.8 CWE 362 (RC)

In this flaw, concurrent operations are executed on a single resource without any proper synchronization. The code requires that certain states should not be modified between two operations, but a timing window exists in which the state can be modified by an unexpected actor or process. Such conditions allow a remote user to take advantage of the race

by executing a series of commands and cause DoS (Denial of Service) attack. Race condition occurred when simultaneously two different codes were executed and access the same memory. It is violated by two properties: Exclusivity and Atomicity. In Exclusivity, until the execution of the current sequence is finished, no other sequences of code can be executed or modify the shared resource properties. Whereas in Atomicity, two or more processes are not executed the same instruction sequence concurrently at the same time [125].

Recently (2021-07-14), a remote user attack on FortiSandbox software. The remote attacker uses race condition in FortiSandbox's command shell and performs DoS attack by executing a series of commands and its impact was medium [126].

3.2.9 CWE 454 (EITV)

The critical internal variables are initialized by software or stored data by using inputs fields that can be manipulated by unauthorized users (Or the software uses inputs that can be changed by untrustworthy actors to initialize critical internal variables or data stores). If any variables have been externally initialized, they should be distrusted, specifically in the case of users because there is the possibility of incorrect initialization. As the result of improper initialization of variables may abrupt the software response and raise the vulnerability in software security. It may allow an attacker to set the variable value and they can easily control what the susceptible system does [127].

Example- This code looks for a debug switch in an "HTTP POST" request and activates debug mode if one is found.

```
$debugEnable = false;
if ($_POST["debug"] == "true"){
$debugEnable = true;
}
```

```
/.../
function Userlogin($userid $userpwd){
if($debugEnable){
echo 'Debug is Activated';
php_info();
$is_Admin = True;
return True;
}
}
```

To enable debug mode, any user with administrator access can activate it, and using the php_info() method information can be printed, which may provide access to an attacker to further exploit the system [128].

3.2.10 CWE 915 (ICMD)

This vulnerability occurred when software used an upstream component (client to server) to receive input data that defines several variables, fields, or properties in an object that should be updated or initialize. But it is unable to control appropriately that which attributes may be modified. If any attributes of an object were only solely meant for internal use, then its unintentional modification may result in a security flaw. Sometimes this weakness is known as language-specific mechanisms [129].

3.3 Security Attributes

Security is a multi-dimensional and comprehensive process that involves a large gamut of operations divided into several stages to ensure in-debt analysis of security-related challenges and threats and ways to mitigate the problems that could affect the operations of a healthcare web application system. The five-set attributes are confidentiality, integrity,

availability, and access control are shown in figure 3.1. These attributes form the basic fundamentals of security; without them the security of software cannot be ensured. The main reason for using these attributes is to plug in gaps in the healthcare web application structure so that security breaches could not be made [130].

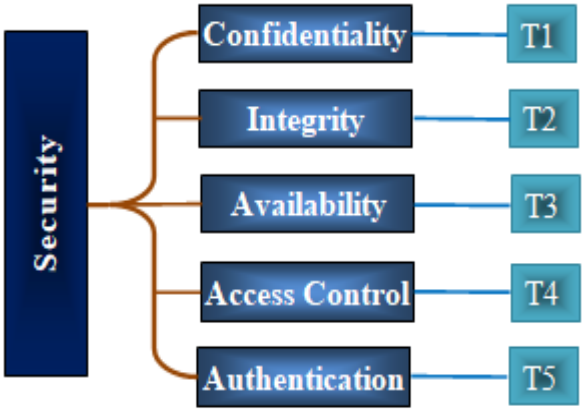

Figure 3.1: Security Attributes

3.3.1 Confidentiality

Confidentiality ensures that the data is not disclosed to any unauthorized user. Inability to maintain confidentiality can lead to data breaches and the leak of sensitive data to unauthorized persons. It ensures that a certain degree of privileges to registered/authorized persons who are permitted to access, make changes and download information and this privilege is restricted or not granted to the unauthorized user/personnel [131]. Confidentiality is a broad security concept that is implemented at all stages of the operation of a software system–processing, storage, retrieval and display of information.

3.3.2 Integrity

Integrity is the accuracy of the data at storage or during transmission. It assures that the end-user's data does not get corrupted or tampered during the transmission can be defined as the accuracy of

data/information at storage or during transmission. In a more expanded form, integrity can be ensured both at the source and destination, which can prevent the unauthorized use of data [132].

3.3.3 Availability

Availability attribute ensures that a system is ready and available for use by an authorized user whenever needed. The availability of a system may be compromised in case of a denial-of-service attack [133]. The availability of a system confirms that the system is ready to be used to all the needed functionalities. The system should be designed in multiple subsystems so that the availability of the system is not jeopardized in case of the failure of any of the subsystems.

3.3.4 Access Control

Access control limits the way the system should be used by its legitimate users. The users are required to present credentials in order to access the specific functionality of the system. The users are decided into levels based on their access controls. Some of the users may be given full control of the system like the administrators, while other users may be given only limited access like the end-users based on their specific use of the system [134].

3.3.5 Authentication

Authentication is the process of identifying the legitimate user requesting access to the system. A user name and password are the most common method of authenticating any user to provide access. The process of authentication involves a mechanism which validates authentic users or multiple users to access information. This authentication can be in the form of a security question, SMS, OTP, biometric, and RSA etc. [131].

3.4 Mapping

Healthcare web application security is an important issue/concern, which makes sure that the application is reliable and secure. The attributes of software security are correlated with security risk factors. The security is purely dependent on all the parameters of the security plan specification [105].

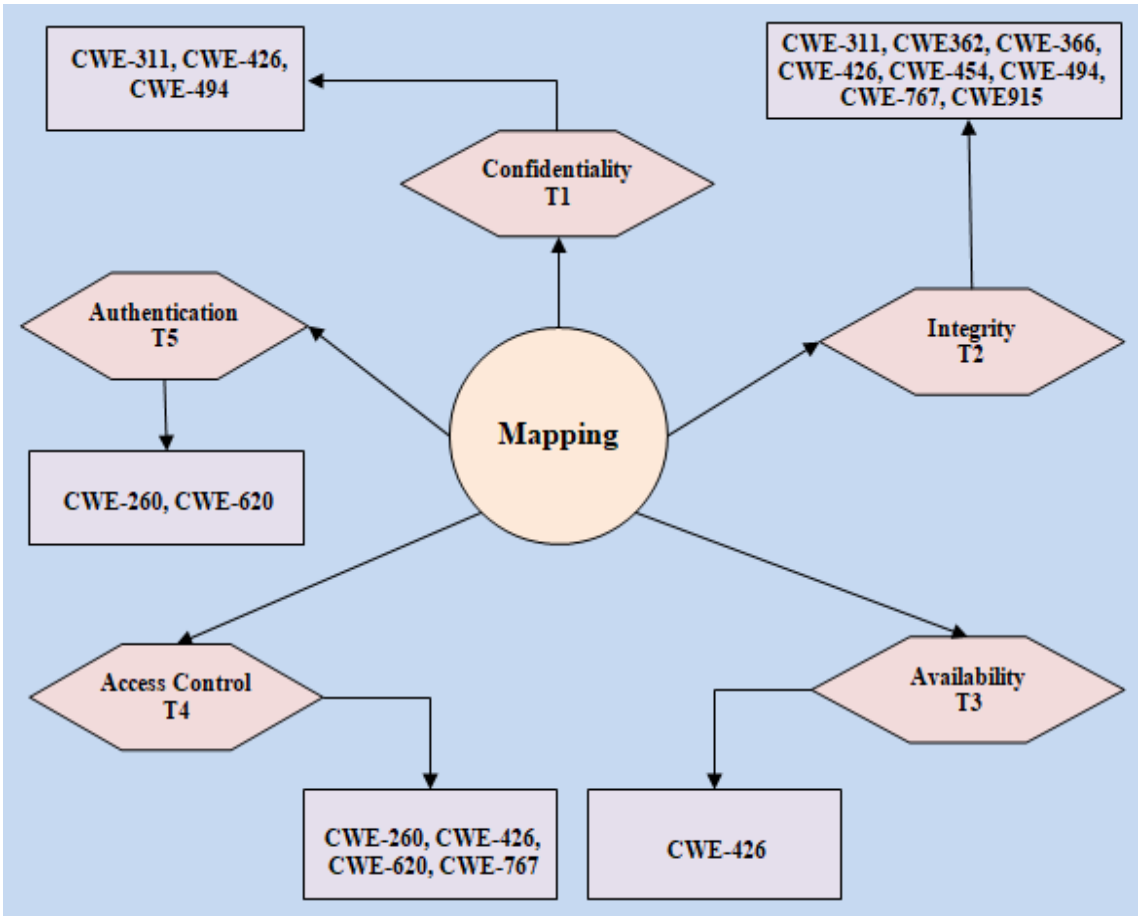

Figure 3.2: Mapping between Security Risk Factors and Security Attributes

It is clear from the literature survey that all security attributes are parts of the security plan specification and play a vital role in security estimation. In figure 3.2, it can be seen how these attributes are correlated to security risk factors. With the help of that, developers may conclude facts regarding the software's security.

3.5 Conclusion

Software security is the branch of software engineering that aims to prevent the exploitation of security loopholes in the system and detect possible vulnerabilities that may prove harmful to the software. For building software systems without any security ambiguity, it is essential to take major security attributes in the development process, failing which can lead to damage to the software system's integrity. The proper use and implementation of security attributes can accelerate the process of finding vulnerabilities & flaws in the system and their adequate mitigation strategy. The successful implementation of a security plan may converge the developing team's entire focus to select a few errors/vulnerabilities that may have affected the healthcare web application system, and they can prepare a strategy for a timely recovery.

In this chapter, a mapping between security attributes and security risk factors is created on behalf of the literature survey and from the expert's view, which was already described in the previous chapter. The use of security attributes in the development lifecycle comes under the ambit of security plan specifications at various stages, without which the security of the software system cannot be insured. These security attributes will help in the optimization, which will further help us understand the types of threats that a particular system can go through. Since each healthcare web application has a particular purpose and performs a particular set of functions according to the need of the organization/user. Hence, every security scenario will be different from others, but these attributes will be included in preparing a roadmap for security strategy in the healthcare web application systems.

Chapter 4

A UNIFIED SECURITY RISK FRAMEWORK FOR HEALTHCARE WEB APPLICATION

4.1 Introduction

Nowadays, human life is much more dependent on the software system that directly increases pressure on development teams and researchers. The main focus of these teams is on securing software systems from vulnerabilities and threats. Reducing vulnerabilities are always directly proportional to security. Through the study of available statistics, it is observed that security has played a vital role in the success as well as failure of software systems.

Based on the recent study, software companies are considering to introduce a software security risk framework in the early phase of development and not just depending on the later phase, when the product is finalized. This step could improve the situation and reduce losses to a substantial level. An estimated loss of more than USD 100,000 has been incurred by a company because of cybercrime, as reported by the Cyber Crime Website of the Department of Justice, USA [135]. This is one of the infinite numbers of cases worldwide where insecure software systems are breached and exploited by attackers for financial and other gains. According to a report given by the Computer Crime and Intellectual Property Section (CCIPS) of the US Department of Justice, organized multinational criminal enterprises have arisen that specifically target software systems and exploit its weaknesses to salvage financial information [136]. In recent years, threats from botnets (network of surreptitiously infected computer systems due to malware) have increased drastically.

Cybercriminals purchase access to botnets and use the network of infected computer systems for various crimes, usually financial data thefts, dissemination of spams, concealing other crimes or Distributed Denial of Service (DDoS). As per a white paper published by Cyber Unit CCIPS, US Department of Justice, many public and private organizations are increasingly adopting vulnerability disclosure programs, which increase their ability to detect security issues and protect sensitive data and prevent disruption of services [137]. A built-in software security framework that includes all security attributes can be a viable and potent solution to numerous security issues. It can prove to be a boon to the users/organizations/governments spending billions of dollars every year on securing their networks.

The recent trends show that healthcare web application security is gaining attention from industries and researchers. As a solution to these problems, developers should try to build an end-to-end healthcare web application security risk assessment framework to detect, assess and mitigate the risks. To facilitate, the developers have proposed a framework for the assessment of vulnerabilities by applying a security risk framework to improve the security of healthcare web applications. Further, the developers have proposed and used soft computing techniques for designing the framework and given some key activities required to integrate security into software design. The researcher has made an effort to develop some important recommendations and set some basic criteria for creating the guidelines to facilitate the development process for creating a secure healthcare web application, which can enhance security and examine vulnerabilities during the design phase of the healthcare web application system.

The main aim of the security risk assessment framework is to cover all the security breaches that occur during the development of the healthcare web application and after the deployment of the healthcare web application. It will also ensure that the healthcare web applications are protected against real-time attacks in a live environment [138]. Organizations wish to design secure healthcare web applications to gain the trust of end-users and get user satisfaction. Unfortunately, a faster development rate and lack of proper documentation in software development have made security breaches that affect the time and cost of the software [105, 139]. Literature survey shows that the existing methodologies to develop secure healthcare web applications are theoretical or just best practices. Most organizations that aim to achieve healthcare web application security cannot cover all the aspects through which vulnerability occurs; hence, some aspects are ignored. However, focusing on the healthcare web application security framework simultaneously will satisfy the end user's security needs by including greater security, which incorporates all security parameters to create a robust & secure healthcare web application.

Mainly, all the experts of the industries have focused on the deployment phase of the healthcare web application development for improving security longevity and minimizing maintenance cost and time. Hence, integrating security at the design phase will reduce the time span and cost of software development by reducing the development team's rework [139-140]. However, practitioners and researchers have suggested to integrate security during the design phase, but none have taken under consideration the end-to-end security framework to quantify the risks and find optimum measures to mitigate them. In addition, this approach will provide a step-by-step procedure to integrate it with the design phase to improve the healthcare web application security. The entire security process is accompanied by a security risk assessment plan based on

security attributes like confidentiality, integrity, availability, access control, and dissipating those vulnerabilities that may affect the entire project [141]. To provide reliable and secure healthcare web applications and gain user's trust, it becomes our necessity to develop a security framework that provides complete guidance for integrating security at the design phase of healthcare web applications.

4.2 Importance of Security Risk Assessment

Security is an idea that is implemented to protect an application from malicious as well as from various risks [142-143]. As we know that individuals and organization are heavily dependent on software for their daily activities. The demand for security is growing day by day. Hence, the more accuracy ensured during security, and the lesser will be the occurrence of security risk. A robust security framework can help to uncover entire security issues as well as privacy issues by mitigating security risk factors. In addition, it can ensure the security of healthcare web application from illegal access, malicious attacks and other forms of dangerous vulnerabilities. The following are some of the top security threats as per OWASP [144]:

- Using Known Vulnerable Components
- Security Misconfiguration
- Broken Authentication & Session Management
- Sensitive Data Exposure
- Injection
- Cross-Site Scripting
- Missing Function Level Access Control
- Invalidated Forwards and Redirects

Considering the above-mentioned security threats, a conclusion can be drawn that an effort should be made to address these security issues for managing efficient, secure healthcare web application development.

4.3 Proposed Framework

The design phase is the backbone of any software system irrespective of its nature and area of use [145]. Software development organizations have shown enormous growth in developing more and more secure healthcare web applications, but today's urgent need is to detect vulnerabilities at the very initial design phase by an appropriate security risk assessment framework. Through the implementation of the framework, early detection of vulnerabilities can be performed, which will save a considerable amount of time and production cost. The framework works on five phases, i.e., Factors Identification, Mapping, Assessment, Statistical Analysis and Review and Revision will be undertaken when required. The forthcoming section will describe the various phases of the healthcare web application framework. Furthermore, we have developed a conceptual framework that has enlightened the key activities to be accomplished with the intention to increase security at the design of the healthcare web application, as shown in the figure: 4.1.

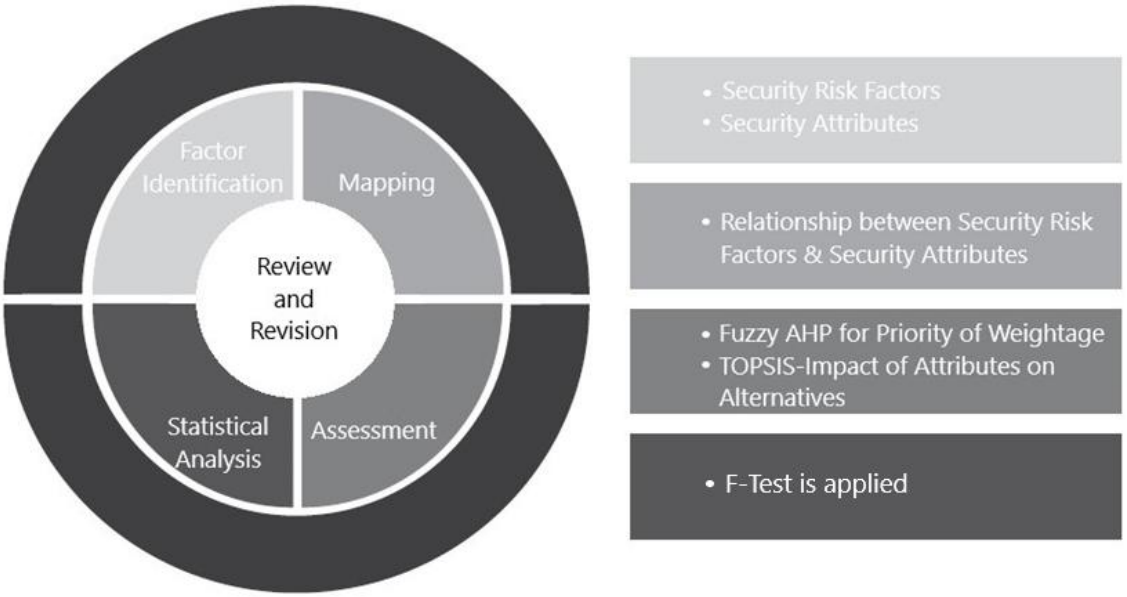

Figure-4.1: Proposed Framework for Securing Healthcare Web Application

4.3.1 Factors Identification

Identification of security risk factors plays a key role in security estimation as well as it creates a roadmap for security professionals for the development of secure healthcare web applications. In this section, an effort has been made to identify the security risk factors and security attributes that may help to quantify design. The aim of this identification is to target key risk factors at the design phase at mitigate them at the earlier phase of the development life cycle. This can be done by quantifying security factors, which will give the developers an advantage of early detection of vulnerabilities in the healthcare web application system and keep know-how of the number of vulnerabilities at any stage in the entire development process. Best results could be achieved by considering all the factors related to security when creating a security estimation plan. This can help in improving the quality of healthcare web applications with the help of an early detection mechanism.

4.3.2 Mapping

In this section, attributes of software security are correlated with security risk factors. It is clear from the literature survey that all security attributes play a vital role in security estimation. This can be done through an in-depth literature survey and expert's points of view. It gives developers an understanding and overview, i.e., whether the security requirements are fulfilled or not. Figure: 3.2 shows a detailed view of mapping between security risk factors and their corresponding security attributes.

4.3.3 Assessment

This section explains how the security risk is being assessed through the integrated technique, i.e., fuzzy AHP-TOPSIS. The author has used fuzzy AHP for the prioritization of security risk factors, and on the other hand with the help of fuzzy AHP-TOPSIS, the impact of attributes

on different alternatives has been calculated. Security risk factors will be prioritized corresponding to their respective weights and ranked accordingly to take a case-by-case security procedure and find out loopholes and other vulnerabilities in the software. Furthermore, this will also help the developers to develop mitigation techniques for the potential threats and guide them to concentrate on those parts of the system on which attackers may influence. The identification and prioritization of the risk factors will provide a path to develop a secure healthcare web application.

4.3.4 Statistical Analysis

Statistical analysis is a mathematical practice that includes gathering of data, organizing it, analysis, and finally drawing an inference from the interpretation of numerical data. For the purpose of explaining the statistical significance of the proposed framework, statistical analysis has been carried out on the alternatives (i.e., healthcare web application) obtained from SAQ Infosys, Lucknow.

4.3.5 Review and Revision

The review & revision may be done, if required, at all stages to bring forth more refined structural guidelines.

4.4 Limitations of the Framework

The framework is a hypothetical description of a very complex yet evolving process that provides a platform for future research work, but every framework has its limitations. The following are the limitations of this proposed framework:

- The commonly accepted factors for security are Confidentiality, Integrity, Availability, Access Control and Authentication is only considered in the proposed framework.

- Since the industrial data is unavailable, the proposed framework has been given to a small set of data.
- This approach is only applicable to the early stage, i.e., at the design phase of an application.

4.5 Conclusion

A detailed study of literature review has shown so far, only a few researchers have cognate security attributes as well as security risk factors to their respective weightage and ranking through the MCDM approach. The propounded framework given in this chapter has come up with the idea of associating security risk factors with weightages and further ranking them. On account of literature reassessment and observing the vibrant scenario & needs of industries today, it is found that security must be capable enough to counter all sorts of vulnerabilities and future threats. This can be achieved by including a desirable security framework to eliminate or reduce vulnerabilities using the above "given security apparatus" at the initial design stage.

It is imperative to understand the users/organizations security needs as they can be complex in nature, without which it is highly unlikely to design a user-specific security framework for the healthcare web application. The structural guidelines may be modified according to the need of the hour to concatenate the best of security attributes and create a secure healthcare web application that satisfied the organization's needs. Further, the framework emphasizes on the optimization of security risk, which will help to save a considerable amount of production cost and time in the development of a healthcare web application. Implementation of the proposed framework gives a superior security detection rate as compared to traditional techniques.

Chapter 5

NUMERICAL ANALYSIS AND INTERPRETATION

5.1 Introduction

Security is an ever-expanding field. The development process for secure healthcare web application is a tenacious task that includes analysis of all the security properties [146]. The modus operandi of security framework includes five attributes that include authentication, access control, availability, integrity, confidentiality. These attributes are a keystone to identify the vulnerabilities and to mitigate them [105]. Software has become one of the most important requirement of users in all the sectors. Hence, its security correspondingly shows its importance and weightage that covers overall purport which can increase or decrease the value of the organization. So, it becomes the developer's necessity to find newer techniques and adopt appropriate measures.

Security is constantly growing and it includes the advanced technologies which can be used for the development of secure healthcare web applications. Intruders are consistently preparing themselves to attack both system and user applications, which might or might not be able, lacerate the security of the software easily [147]. The main aim of intruders is to exploit the weakness of application and acquire sensitive information through which intruders can take control of system.

Hence, to avoid all the possible disruption proposed security framework provides a platform via which its behaviour may be effortlessly controlled. To improve the security of any application, an enactment of security plays an important role. The proposed security framework uses a structured outlook throughout the SDLC, particularly during the design phase. It gives a better sightedness of the application

and keeps it safe against investigated security threats that may arise at any time [105, 148]. At the same time, the number of issues or incidents related to security is growing, and it is becoming a big concern among business owners and IT professionals. Those organizations that do not focus on security and its related factors during each phase of development, the application delivered by them may have hazardous vulnerabilities that may create enormous risks to the organization.

The fundamental objective is to find the priorities which are based on ranking and weightage of the security attributes with the help of MCDM (i.e., Multi-Criteria Decision Making) process under which the exercise of the analytical hierarchical process is shown, through which the application becomes more and more secure and reliable. Since no attempts have been made to quantitatively prioritize and rank the security attributes that may affect the success of software security and their trade-offs, the fuzzy AHP approach can be utilized to prioritize the security attributes in terms of well-profiling. This improves the life span of software and early detection of vulnerabilities which will directly benefit users/organizations by enhancing the security of software. The analysis of prioritization of security attributes by using the Analytic Hierarchical Process, which is a type of MCDA (i.e., Multi-Criteria Decision Analysis) [149, 150]. Multi-Criteria Decision Analysis is helpful for performing various evaluations of conflicting elements like Analytic Hierarchical Process and Multi-Attribute Utility Theory [151]. Three distinct parts of the MCDA process are objectives, alternative weights and their ranks which are used in the AHP technique.

Analytical Hierarchical Process (AHP) is recognized righteous in analysing a conclusion in the group. In addition, it has been observed that in case of fuzzy AHP more accurate relationship has been established, which provide complete priority analysis by decision-makers [152].

Hence, to construct a network of attributes according to their importance or priority, fuzzy AHP works with judgmental input from a group of decision-makers. This study establishes a foundation for assessing security by using a fuzzy AHP approach. Here researchers have conglomerated data from security experts of different fields of academics and industry. The aim is to evaluate the security risk factors in terms of their weight and ranks. On behalf of these results, security development strategies are selected to mitigate these risk factors.

5.2 Estimation Mechanism

In the current study, the researcher has used an integrated technique, i.e., fuzzy AHP-TOPSIS for the assessment of security.

5.2.1 Integrated Fuzzy AHP-TOPSIS Method

Numerous researchers have accomplished research on the basis of security. Current promulgation for high-end security is transforming security of healthcare web applications [153]. Furthermore, challenges with MCGDM (i.e., Multi-Criteria Group Decision Making) are common in exercise for attaining the goals, based on user's need and with respect to the sensitivity of information. There are various approaches in the literature that can be used to overcome such issues [154]. The AHP approach is preferable to other MCDA processes for accessing the values (i.e., subjective and objective) of variables. However, this approach is unable to overcome the inherent incertitude as well as the ambiguity of a response from decision-maker to precise statistics.

The researcher of this perusal discovered that specialists have used fuzzy theory along with AHP to examine the imprecise real-world's challenges because it is exceedingly ambiguous [155]. In addition, the AHP approach is rooted on unsteady scale; on the other hand the fuzzy AHP approach has the same flaws [154-155]. Hence, as a result a unique

prescript (i.e., a unified fuzzy perspective of AHP and TOPSIS) for systematically evaluating alternatives based on multiple criteria.

5.2.2 Fuzzy AHP

Fuzzy AHP is a powerful prescript for addressing arduous conclusive problems, and all the complicated problems may be evaluated through various classed levels of objectives. To solve the arduousness of a complex problem, fuzzy AHP divides it into a tree-like structure. In addition, for the estimation of priority of various alternatives with multiple criteria in a hierarchical structure, it is also utilized as a decision-making technique/tool [156]. The fuzzy AHP is based the fuzzy interval arithmetic, which uses the TFN to compute the weights of elements. Saaty was the first who proposed the AHP technique [157]. To deal with imprecision in multi-criteria decision problems, it merely uses the pair-wise comparison matrix [158]. The triangular fuzzy numbers are used in this model to represent linguistic variables and to conduct fuzzy operations using AHP. To deal with the uncertainty caused by imprecision and vagueness, Zadeh has developed the fuzzy set theory [159].

On the basis of expert's viewpoints as well as responses via questionnaire or by using brainstorming, the tree structure is prepared, and after that Triangular Fuzzy Number (TFN) is fabricated from the hierarchy. In addition, pair-wise comparison of every cluster of grouped objective performs a significant contribution in determining the impact of one criterion on the other. The researcher transfigures the linguistic values into TFN as well as crisp numbers, and in this study the values of TFN are between 0 to 1 [160]. The computational simplicity of triangular fuzzy membership functions as well as their ability to deal/cope with fuzzy data is the reason for their widespread acceptance [154].

Additionally, the classification of linguistic values is as equally important, fairly important, strongly important, weakly important, and absolutely important etc., and apart from these the crisp values are grouped as 1,2, ….9. Furthermore, a fuzzy number T on F is called TFN, if its membership functions are depicted in equations (1-2):

$$\mu_a(t) = \mathrm{F} \rightarrow [0,1] \quad (1)$$

$$\mu_a(t) = \begin{cases} \dfrac{t}{m_i - l_o} - \dfrac{l_o}{m_i - l_o} & t \in [l_o, m_i] \\ \dfrac{t}{m_i - u_p} - \dfrac{u_o}{m_i - u_p} & t \in [m_i, u_p] \\ Otherwise, \quad \mu_a(t) = 0 \end{cases} \quad (2)$$

Here, in the triangular membership function l_o, m_i, and u_p are given as limit (i.e., lower limit, middle limit, and upper limit, respectively).

A TFN is shown in Figure 5.1.

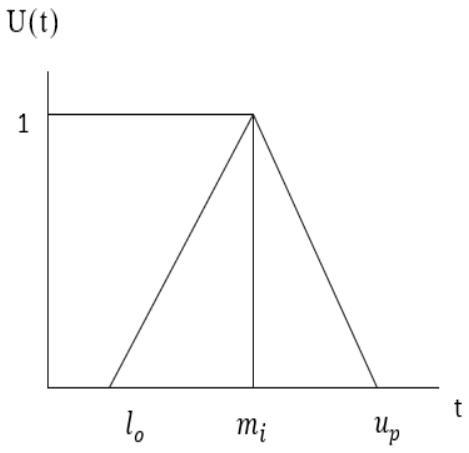

Figure 5.1: Triangular Fuzzy Numbers

(l_o, m_i, u_p) is depicted as TFN. In a quantitative manner, experts have assigned ratings to the factors affecting the values using the scale shown in Table 5.1.

TABLE 5.1: TFN Scale

Saaty Scale Definition	Fuzzy Triangle Scale	
1	Equally important	(1, 1, 1)
3	Weakly important	(2,3, 4)
5	Fairly important	(4,5, 6)
7	Strongly important	(6,7, 8)
9	Absolutely important	(9,9, 9)
2	Intermittent values between two adjacent scales	(1,2, 3)
4		(3,4, 5)
6		(5,6, 7)
8		(7,8, 9)

In the conversion of numeric values into TFN, equations (3-6) are used [154-155,160] that are designated as $(l_{o_{jk}}, m_{i_{jk}}, u_{p_{jk}})$ where, $l_{o_{jk}}$ is lower value, $m_{i_{jk}}$ is middle value and $u_{p_{jk}}$ is uppermost value. Additional, TFN $[n_{jk}]$ is recognized as:

$$n_{jk} = (l_{o_{jk}}, m_{i_{jk}}, u_{p_{jk}}) \quad (3)$$

Where, $l_{o_{jk}} \leq m_{i_{jk}} \leq u_{p_{jk}}$

$$l_{o_{jk}} = \min (R_{jkz}) \quad (4)$$

$$m_{i_{jk}} = (R_{jk1}, R_{jk2}, R_{jk3})^{\frac{1}{t}} \quad (5)$$

$$and \quad u_{p_{jk}} = \max (R_{jkz}) \quad (6)$$

In the equations (3-6), R_{jkz} indicates the relative importance of the values between two factors which is given by security experts z, where j and k signify a pair of factors being decided by security experts. n_{jk} is evaluated for a specific comparison on the basis of geometric mean, which is given by experts. The geometric mean is capable of appropriately integrating as well as representing practitioners consensus, and it denotes the scores (i.e., lowest and highest) for the relative weightage of two factors. Additionally, equations 7, 8, and 9 used to combine TFN values. Consider two TFNs T1 and T2, T1= $l_{o_1}, m_{i_1}, u_{p_1}$ and T2= $l_{o_2}, m_{i_2}, u_{p_2}$. The rules of operations on them are as:

$$\left(l_{o_1}, m_{i_1}, u_{p_1}\right) + \left(l_{o_2}, m_{i_2}, u_{p_2}\right) = (l_{o_1} + l_{o_2},\ m_{i_1} + m_{i_2}, u_{p_1} + u_{p_2}) \quad (7)$$

$$\left(l_{o_1}, m_{i_1}, u_{p_1}\right) \times \left(l_{o_2}, m_{i_2}, u_{p_2}\right) = (l_{o_1} \times l_{o_2},\ m_{i_1} \times m_{i_2}, u_{p_1} \times u_{p_2}) \quad (8)$$

$$\left(l_{o_1}, m_{i_1}, u_{p_1}\right)^{-1} = \frac{1}{u_{p_1}}, \frac{1}{m_{i_1}}, \frac{1}{l_{o_1}} \quad (9)$$

$$\left(l_{o_1}, m_{i_1}, u_{p_1}\right) + \left(l_{o_2}, m_{i_2}, u_{p_2}\right) = (l_{o_1} + l_{o_2},\ m_{i_1} + m_{i_2}, u_{p_1} + u_{p_2})$$

With the help of equation 10, a fuzzy pair-wise comparison matrix in the form of $m \times m$ matrix is generated after obtaining the TFN values for each pair of comparisons.

$$\widetilde{P^z} = \begin{bmatrix} \tilde{x}_{11}^z & \tilde{x}_{12}^z \ldots\ldots & \tilde{x}_{1m}^z \\ \tilde{x}_{21}^z & \tilde{x}_{22}^z \ldots\ldots & \tilde{x}_{2m}^z \\ \ldots\ldots & \ldots\ldots & \ldots\ldots \\ \tilde{x}_{m1}^z & \tilde{x}_{m2}^z \ldots\ldots & \tilde{x}_{mm}^z \end{bmatrix} \quad (10)$$

Where, $\widetilde{x_{jk}^z}$ represents the z^{th} decision maker's preference of the j^{th} criteria over the k^{th} criteria. When there are multiple decision-makers equation (11) is used to calculate the average of each decision maker's preferences.

$$\widetilde{x_{jk}} = \sum_1^d \tilde{x}_{jk}^z \quad (11)$$

After that, with the help of equation (12) and based on average preferences, pair-wise comparison matrices are updated for all the factors in the hierarchy.

$$\tilde{P} = \begin{bmatrix} \widetilde{x_{11}} & \cdots & \widetilde{x_{1m}} \\ \vdots & \ddots & \vdots \\ \widetilde{x_{m1}} & \cdots & \widetilde{x_{mm}} \end{bmatrix} \quad (12)$$

The fuzzy geometrical mean and fuzzy weights of each factor are then described using the geometrical mean technique, as indicated in equation (13).

$$\widetilde{a_\iota} = \left(\prod_{i=1}^{n} \widetilde{x_{jk}}\right)^{1}/_{n} \ , j = 1,2,3, \ldots . n \quad (13)$$

After that, with the help of equation (14) fuzzy weight of the factor is concluded.

$$\widetilde{wt_\iota} = \widetilde{a_\iota} \otimes (\widetilde{a_1} \oplus \widetilde{a_2} \oplus \widetilde{a_3} \ldots\ldots \oplus \widetilde{a_n})^{-1} \quad (14)$$

Further, with the help of equations (15-16), the average and normalized weight criteria are to be calculated.

$$Avg_i = \frac{\widetilde{wt_1} \oplus \widetilde{wt_2} \ldots\ldots \oplus \widetilde{wt_n}}{n} \quad (15)$$

$$N\_wt_i = \frac{Avg_i}{Avg_1 \oplus Avg_2 \oplus \ldots\ldots \oplus Avg_n} \quad (16)$$

Furthermore, to compute the BNP value of the fuzzy weights the Centre of Area (COA) approach is applied for every measurement with the help of equation (17).

$$BNP_{Wt} = \frac{[(\mathrm{u_p} wt1 - l_o wt1) + (m_i wt1 - l_o wt1)]}{3} + l_o wt1 \quad (17)$$

5.2.3 Fuzzy TOPSIS

TOPSIS consider a multi-criteria decision-making view with m choices as a geometric arrangement with m points in the n-dimensional space of factors. The method utilized in this study is based on the assumption that for maximum and minimum ideal solutions, respectively, a selected alternative has the shortest and farthest distance from the positive-ideal solution and negative-ideal solution [161]. Shadbegian and Gray stated that security experts might encounter some issues for the allocation of specific performance ratings of any alternative on the basis of factors. Step-by-step process of Fuzzy AHP-TOPSIS method is explained in the given flow chart.

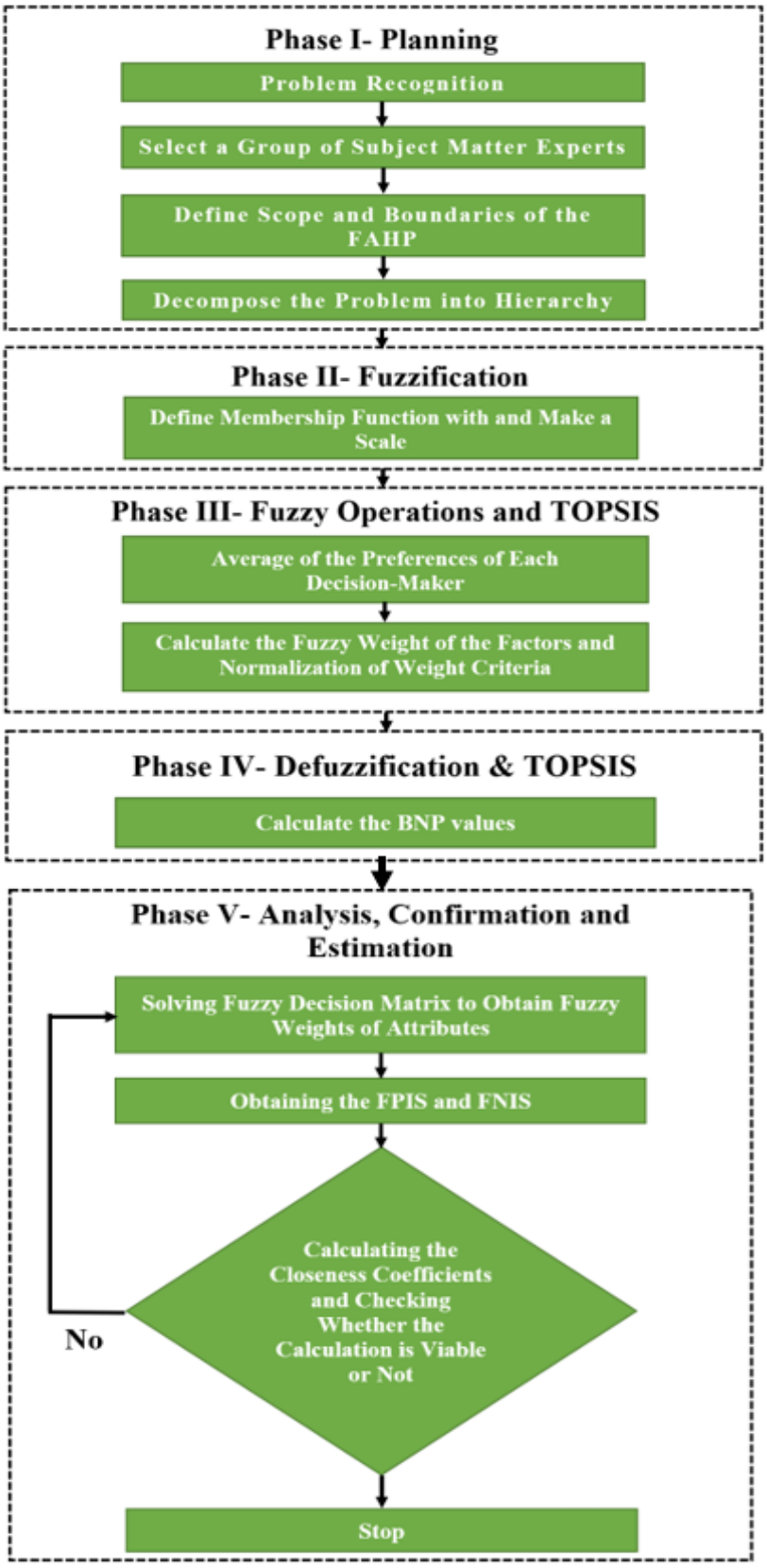

Figure 5.2: Flow Chart of Fuzzy AHP-TOPSIS Method

This procedure allocates fuzzy numbers in place of specific numbers to represent the relative significance of a factor for consistency with real-world fuzzy surroundings. Furthermore, the fuzzy AHP-TOPSIS technique is well suited to solve group decision-making problems in fuzzy contexts. Figure 5.2 illustrates the comprehensive procedure of achieving weights as well as for the estimation of the viability of the fuzzy AHP-TOPSIS method.

Firstly, the researcher determines the weights of the evaluation factors. With the help of equations (1-16), the current research applies fuzzy AHP process to derive fuzzy weight. In addition, a fuzzy decision matrix is created by researchers with the help of table 5.2 and equation (18), and relevant linguistic variables are chosen as alternatives for the criterion.

TABLE 5.2: Linguistic Scales for the Rating

Linguistic Variable	Corresponding Triangular Fuzzy Number
Very poor (VP)	(0, 1, 3)
Poor (P)	(1, 3, 5)
Fair (F)	(3, 5, 7)
Good (G)	(5, 7, 9)
Very good (VG)	(7, 9,10)

$$\tilde{X} = \begin{matrix} P_1 \\ \vdots \\ P_m \end{matrix} \begin{matrix} Q_1 & \cdots\cdots & Q_n \end{matrix} \begin{bmatrix} \widetilde{y_{11}} & \cdots & \widetilde{y_{1n}} \\ \vdots & \ddots & \vdots \\ \widetilde{y_{m1}} & \cdots & \widetilde{y_{mn}} \end{bmatrix} \qquad (18)$$

Where, $\tilde{y}_{jk} = \frac{1}{z}(\tilde{y}^1_{jk} \ldots \oplus \tilde{y}^z_{jk} \oplus \ldots\ldots \tilde{y}^Z_{jk})$, and $\tilde{y}^z_{jk}$ is the performance rating of the alternative P_j with respect to factor Q_k estimated by the z^{th} practitioner and $\tilde{y}^z_{jk} = ({l_0}^z_{jk}, {m_i}^z_{jk}, {u_p}^z_{jk})$. With the help of equation (19), the fuzzy decision matrix is normalized and it is represented by $\widetilde{D}$.

$$\widetilde{D} = \left[\tilde{a}_{jk}\right]_{m\times n} \quad (19)$$

After that, with the help of equation (20), the normalization process can be achieved.

$$\tilde{a}_{jk} = \left(\frac{l_{o_{jk}}}{u_{p_k^+}}, \frac{m_{i_{jk}}}{u_{p_k^+}}, \frac{u_{p_{jk}}}{u_{p_k^+}}\right), u_{p_k}^+ = \max\{u_{p_{jk}}, j = 1,2,3,\dots,n\} \quad (20)$$

Alternatively, we can set the best-desired level $u_{p_k}^+$ and k = 1, 2, 3, 4,…n is equal to 1; otherwise, the worst is 0. The normalized $\tilde{a}_{jk}$ continues to be TFNs. For trapezoidal fuzzy numbers, the normalization process can be performed in the similar manner. The decision matrix$(\widetilde{D}_w)$ is normalized by weighted fuzzy numbers and is quantified with the help of equation (21).

$$\widetilde{D}_w = \left[\tilde{b}_{jk}\right]_{m\times n} \; j = 1,2,3,\dots m; \;\; k = 1,2,3,\dots.,n; \quad (21)$$

Where, $\tilde{b}_{jk} = \tilde{a}_{jk} \otimes \widetilde{wt}_{jk}$ and then, FPIS (i.e., Fuzzy Positive-Ideal Solution) and FNIS (i.e., Fuzzy Negative-Ideal Solution) is described. The $\tilde{b}_{jk}$ are normalized positive TFN, and their ranges belong to the closed interval [0, 1]. Thereafter, researchers can describe the FPIS A_s^+ (aspiration levels) and FNIS A_s^- (the worst levels) as shown in equations (22-23).

$$A_s^+ = (\tilde{b}_1^*, \dots\dots \tilde{b}_k^*, \dots\dots \tilde{b}_n^*) \quad (22)$$

$$A_s^- = (\tilde{b}_1^*, \dots\dots \tilde{b}_k^*, \dots\dots \tilde{b}_n^*) \quad (23)$$

Where, $\widetilde{b}_1^* = (1,1,1) \otimes \widetilde{wt}_{jk} = (Lwt_k,\ Mwt_k,\ Hwt_k)$ and $\tilde{\tilde{b}}_{jk} = (0,0,0),\ k = 1,2,3,4,\dots,n$. For calculating the distance of each alternative from FPIS and FNIS. The distances $(\widetilde{Dis}_j^+ \; and \; \widetilde{Dis}_j^-)$ of each alternative from $A_s^+ \; and \; A_s^-$can

be estimated using the area compensation technique as shown in equations (24-25).

$$\widetilde{Dis}_j^+ = \sum_{k=1}^{n} Dis\left(\tilde{b}_{jk}, \tilde{b}_{jk}^*\right) \; j = 1,2,3,\ldots m; \quad k = 1,2,\ldots.,n; \tag{24}$$

$$\widetilde{Dis}_j^- = \sum_{k=1}^{n} Dis\left(\tilde{b}_{jk}, \tilde{b}_{jk}^*\right) \; j = 1,2,3,\ldots m; \quad k = 1,2,\ldots.,n; \tag{25}$$

In addition, researchers discover the closeness coefficients (i.e., relative gaps–degree) and generated alternatives for the achievement of aspiration levels in each factor.

To improve the alternatives, Ying-Chyi Chou et al. proposed that $Q\tilde{Q}_j$ is cleared to evaluate the fuzzy gaps–degree on the basis of the fuzzy closeness coefficients [160]. The similarities to the ideal solution are determined after evaluating the $\widetilde{Dis}_j^+ \; and \; \widetilde{Dis}_j^-$ of each alternative and is depicted in equation (26).

$$Q\tilde{Q}_j = \frac{\tilde{x}_j^-}{\tilde{x}_j^+ + \tilde{x}_j^-} = 1 - \frac{\tilde{x}_j^+}{\tilde{x}_j^+ + \tilde{x}_j^-} \; , j = 1,2,3,\ldots\ldots,m \tag{26}$$

Where, $\frac{\tilde{x}_j^-}{\tilde{x}_j^+ + \tilde{x}_j^-}$ defined as fuzzy satisfaction degree in the j^{th} alternative and $\frac{\tilde{x}_j^+}{\tilde{x}_j^+ + \tilde{x}_j^-}$ defined as the fuzzy gap–degree in the j^{th} alternative. On the basis of which ranks of alternatives is achieved.

5.3 Empirical Data Analysis and Results

Generally, qualitative assessment seems to be suitable for the assessment of long-term security. It is quite difficult to perform a qualitative assessment of healthcare web application's security. Security strategy is prepared on the basis of results drawn from global collaborative activities. In recent years, security professionals have amassed a large number of security policies [154]. Furthermore, several firms are currently adopting high-end security healthcare web

applications. The impact of security attributes on healthcare web applications plays a crucial role in ensuring security [162]. The researcher of the current study proposes a method for the estimation of security at the design phase, i.e., fuzzy AHP-TOPSIS method.

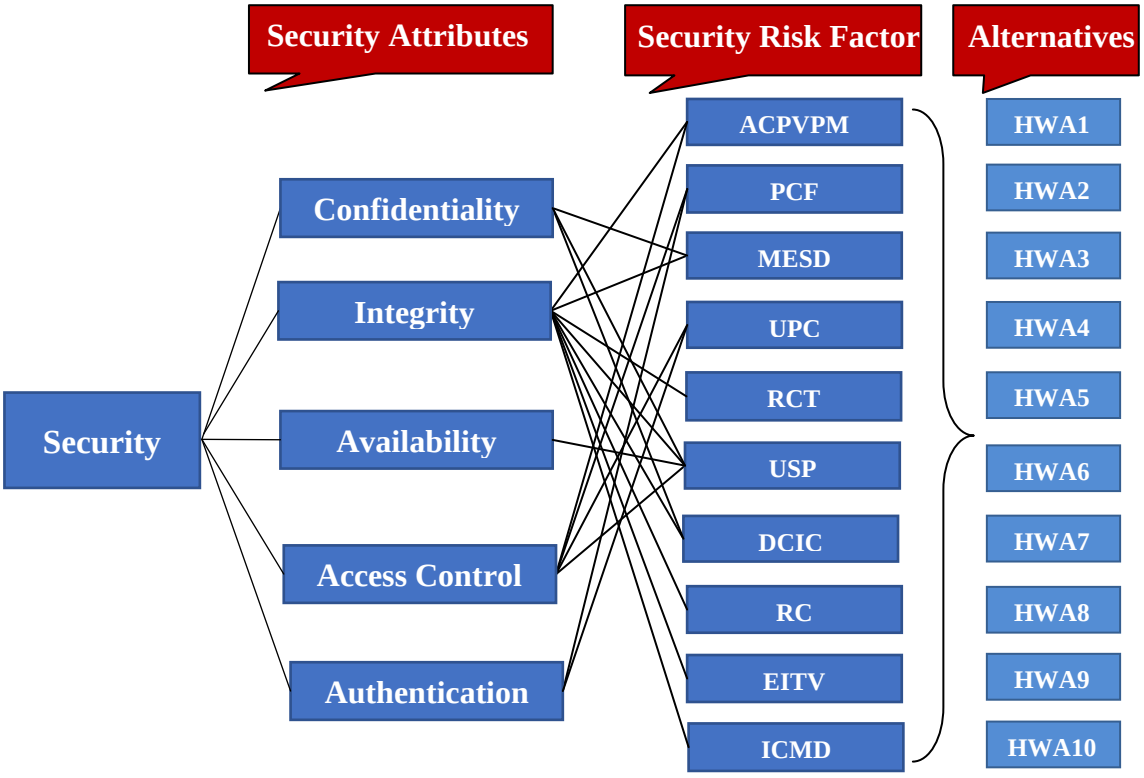

Figure 5.3: A Hierarchy of Security Attributes and Risk Factors

In chapter 3, the researcher has identified various security attributes and risk factors. For the purpose of assessment, the identified security attributes and risk factors are linked together, and the relationship among them is recognized. The hierarchical structure of this process is shown in figure 5.3. For assessment, T11, T12, and T13 are represented as the attributes of confidentiality at level 2 with respect to security. T21, T22, T23, T24, T25, T26, T27, and T28 are represented as the attributes of integrity at level 2 with respect to security. T31is represented as the attributes of availability at level 2 with respect to security. T41, T42, T43, and T44 are represented as the attributes of

access control at level 2 with respect to security. T51 and T52 are represented as the attributes of authentication at level 2 with respect to security.

This study uses the opinions of 70 professionals from academia and industry in order to compile the data. The estimation of security via fuzzy AHP-TOPSIS has been assessed by using equations (i.e., 1-26) as follows:

The researcher has converted the linguistic values into numeric values as well as in aggregated TFNs values by using table 5.1 and equations (1-9). Additionally, equation 10 is used to create the pair-wise comparison matrixes for level 1 attributes is shown in table 5.3. Similarly, table 5.4 - 5.7 shows the fuzzy pair-wise comparison matrixes through the hierarchy at level 1 and 2 respectively.

Table 5.3: Fuzzy Aggregated Pair-Wise Comparison Matrix at Level 1

Level 1	T1	T2	T3	T4	T5
T1	1.000000, 1.000000, 1.000000	0.312700, 0.439500, 0.625200	0.873300, 0.901200, 0.946500	0.226100, 0.292800, 0.416600	0.258000, 0.338600, 0.505500
T2	-	1.000000, 1.000000, 1.000000	2.045100, 3.169900, 4.233000	0.266500, 0.365700, 0.591100	0.690600, 1.005900, 1.511700
T3	-	-	1.000000, 1.000000, 1.000000	0.366701, 0.525102, 0.965903	0.360401, 0.522000, 0.807401
T4	-	-	-	1.000000, 1.000000, 1.000000	0.896002, 1.148603, 1.390300
T5	-	-	-	-	1.000000, 1.000000, 1.000000

Table 5.4: Fuzzy Aggregated Pair-Wise Comparison Matrix at Level 2 for Confidentiality

	T11	T12	T13
T11	1.000000, 1.000000, 1.000000	0.695141, 0.950141, 1.345741	1.104861, 1.438051, 1.690625
T12	-	1.000000, 1.000000, 1.000000	1.190285, 1.582062, 2.149701
T13	-	-	1.000000, 1.000000, 1.000000

Table 5.5: Fuzzy Aggregated Pair-Wise Comparison Matrix at Level 2 for Integrity

	T21	T22	T23	T24	T25	T26	T27	T28
T21	1.0000000, 1.0000000, 1.0000000	1.1121440, 1.5105170, 1.9331210	0.4891060, 0.6301720, 1.5241140	0.410152, 0.574430, 1.652314	0.2210150, 0.2870101, 0.4152130	0.3141440, 0.4611120, 0.8712240	0.6574560, 1.1652352, 1.6882152	0.2442635, 0.3234474, 0.4865241
T22	-	1.0000000, 1.0000000, 1.0000000	0.5704413, 0.6654327, 0.8021522	0.3045749, 0.3934416, 0.5661153	0.2678565, 0.3523652, 0.5175854	0.1668565, 0.1968552, 0.2531145	0.3938857, 0.5745654, 1.0564474	0.1695362, 0.2135264, 0.2751263
T23	-	-	1.0000000, 1.0000000, 1.0000000	1.1141214, 1.3195523, 1.5517541	0.3112564, 0.4311241, 0.8112454	0.8441231, 0.8711241, 1.1253514	1.26112236 1.8245655, 2.4312234	0.17115424, 0.2044512, 0.26411414
T24	-	-	-	1.0000000, 1.0000000, 1.0000000	0.5384547, 0.9147441, 1.5835511	0.6082266, 1.0591177, 1.6828822	0.7545741, 1.3462265, 1.9615522	0.6785697, 0.7474458, 0.8725356
T25	-	-	-	-	1.0000000, 1.0000000, 1.0000000	0.4147584, 0.6344521, 1.1711540	0.9474542, 1.1095110, 1.2457220	0.2511452, 0.3344125, 0.5114411
T26	-	-	-	-	-	1.0000000, 1.0000000, 1.0000000	1.8884521, 2.5515235, 3.1694125	0.8112454, 1.0352748, 1.3166525
T27	-	-	-	-	-	-	1.0000000, 1.0000000, 1.0000000	0.2136856, 0.2574578, 0.3194564
T28	-	-	-	-	-	-	-	1.0000000, 1.0000000, 1.0000000

Table 5.6: Fuzzy Aggregated Pair-Wise Comparison Matrix at Level 2 for Access Control

	T41	T42	T43	T44
T41	1.0000000, 1.0000000, 1.0000000	1.0784541, 1.5991125, 2.1134474	0.8244564, 1.1125424, 1.61447854	0.5674465, 0.713285, 0.8734478
T42	-	1.0000000, 1.0000000, 1.0000000	0.3237858, 0.4488569, 0.6052565	0.2588564, 0.3174658, 0.4164474
T43	-	-	1.0000000, 1.0000000, 1.0000000	0.6667458, 1.0564457, 1.5444547
T44	-	-	-	1.0000000, 1.0000000, 1.0000000

Table 5.7: Fuzzy Aggregated Pair-Wise Comparison Matrix at Level 2 for Authentication

	T51	T52
T51	1.000000, 1.000000, 1.000000	0.666414, 1.050612, 1.542857
T52	-	1.000000, 1.000000, 1.000000

The researcher has calculated the fuzzy weights of factors with the help of equations (11-13). With the help of equations (14-16), the weight of each element is calculated. Additionally, BNP values (i.e., Best Non-fuzzy Performance) of each attribute is calculated with the help of equation 17. Thereafter, the weights for the continuing attributes may be determined and shown in table 5.8 - 5.12, which depict the local and dependent weight of attributes according to figure 5.2. Table 5.13 shows the global weight of every attribute of security.

Table 5.8: Combined Pair-Wise Comparison Matrix at Level 1

	T1	T2	T3	T4	T5	Weights
T1	1.0000000	2.554040	1.701796	2.4274857	0.5909314	0.239131
T2	0.391451	1.0000000	0.796489	0.9706941	0.2070325	0.095012
T3	0.587614	1.255622	1.0000000	1.0563454	0.2532758	0.119936
T4	0.412124	1.023633	0.946789	1.0000000	0.2357854	0.103297
T5	1.66864165	4.82397414	3.94958574	4.2427856	1.0000000	0.442623
						C.R.=0.0025441

Table 5.9: Combined Pair-Wise Comparison Matrix at Level 2 for Confidentiality

	T11	T12	T13	Weights
T11	1.000000	0.985314	1.357784	0.361084
T12	1.014945	1.000000	1.626124	0.387257
T13	0.736541	0.614713	1.000000	0.251659
				C.R.=0.0026120

Table 5.10: Combined Pair-Wise Comparison Matrix at Level 2 for Integrity

	T21	T22	T23	T24	T25	T26	T27	T28	Weights
T21	1.0000000	1.4912124	0.69125650	0.64185470	0.311425	0.52067859	1.1697454	0.3438898	0.073495
T22	0.6741250	1.0000000	0.67785470	0.41044558	0.372411	0.20337758	0.6497445	0.2150884	0.049661
T23	1.4474414	1.4771241	1.00000000	1.29774414	0.493574	0.85022565	1.8364457	0.2147744	0.103253
T24	1.5644144	2.4138541	0.77115640	1.0000000	0.963144	1.10245289	1.3511854	0.7319854	0.127255
T25	3.3141614	2.6852314	2.02638547	1.03788547	1.0000000	0.71724587	1.1028458	0.4358956	0.140713
T26	1.8981151	4.9188151	1.17375840	0.90718564	1.3943544	1.0000000	2.3852744	1.0473857	0.173187
T27	0.8551144	1.5396547	0.54444744	0.7404474	0.9067847	0.41927751	1.0000000	0.2621445	0.076066
T28	2.9154114	4.6484474	4.67295260	1.36634454	2.2989756	0.95487447	3.8153854	1.0000000	0.256370
									C.R.=0.0330124100

Table 5.11: Combined Pair-Wise Comparison Matrix at Level 2 for Access Control

	T41	T42	T43	T44	Weights
T41	1.000000	1.5973121	1.1648124	0.7168241	0.254350
T42	0.6262541	1.000000	0.4561471	0.3274145	0.130199
T43	0.8585152	2.192280	1.000000	1.0804125	0.282926
T44	1.3951125	3.0544412	0.925615	1.000000	0.332525
					CR=0.01870140

Table 5.12: Combined Pair-Wise Comparison Matrix at Level 2 for Authentication

	T41	T42	Weights
T41	1.000000	1.08040	0.519323
T42	0.92560	1.000000	0.480677
			CR= 0.000000

Table 5.13: Final Weights of Hierarchy

Characteristics of Level 1	Local Weights of Level 1	Characteristics of Level 2	Local Weights of Level 2	Global Weights of Level 2
T1	0.239131	T11	0.361084	0.086346378004
		T12	0.387257	0.092605153667
		T13	0.251659	0.060179468329
T2	0.095012	T21	0.073495	0.006982906940
		T22	0.049661	0.004718390932
		T23	0.103253	0.009810274036
		T24	0.127255	0.012090752060
		T25	0.140713	0.013369423556
		T26	0.173187	0.016454843244
		T27	0.076066	0.007227182792
		T28	0.256370	0.024358226440
T3	0.119936	T31	-	0.119936000000
T4	0.103297	T41	0.254350	0.026273591950
		T42	0.130199	0.013449166103
		T43	0.282926	0.029225407022
		T44	0.332525	0.034348834925
T5	0.442623	T51	0.519323	0.229864304229
		T52	0.480677	0.212758695771

Now, the researcher has to figure out the impact of risk factors in alter preferences with respect to criteria. Ten successive healthcare web applications (i.e., HWA1, HWA2, HWA3, HWA4, HWA5, HWA6, HWA7, HWA8, HWA9, and HWA10) from the local hospitals of Uttar Pradesh, India, have been taken to estimate the security risk. The researcher takes input on the technological data with the help of table 5.2 on all 10 alternatives as depicted in table 5.14. The researcher assessed normalized fuzzy decision matrix as shown in table 5.15 by using equations (18-20) and by using equation (21), and evaluated the weighted normalized fuzzy decision matrix as shown in table 5.16. Additionally, the researcher assessed the satisfaction degree and gap degree by using equations (22-26), as depicted in table 5.17.

Table 5.14: Subjective Cognition Results of Evaluators in Linguistic Terms

	HWA1	HWA2	HWA3	HWA4	HWA5	HWA6	HWA7	HWA8	HWA9	HWA10
T11	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500
T12	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100
T13	4.0900, 6.0900, 7.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100
T21	2.4500, 4.2700, 6.2700	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	2.0900, 3.9100, 5.8200
T22	4.0900, 6.0900, 7.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	3.0900, 5.0000, 6.8200
T23	4.8200, 6.8200, 8.5500	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	4.8200, 6.8200, 8.5500
T24	4.0900, 6.0900, 7.9100	4.0900, 6.0900, 7.9100	2.3600, 4.2700, 6.2700	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100	2.3600, 4.2700, 6.2700	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100	2.3600, 4.2700, 6.2700	4.0900, 6.0900, 7.9100
T25	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200
T26	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200
T27	4.0900, 6.0900, 7.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	2.3600, 4.2700, 6.2700

T28	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100
T31	2.4500, 4.2700, 6.2700	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	2.0900, 3.9100, 5.8200
T41	4.0900, 6.0900, 7.9100	4.0900, 6.0900, 7.9100	2.3600, 4.2700, 6.2700	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100	2.3600, 4.2700, 6.2700	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100	2.3600, 4.2700, 6.2700	3.0900, 5.0000, 6.8200
T42	4.8200, 6.8200, 8.5500	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	4.8200, 6.8200, 8.5500
T43	4.0900, 6.0900, 7.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	5.1800, 7.1800, 8.9100	4.8200, 6.8200, 8.5500	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100
T44	5.1800, 7.1800, 8.9100	4.0900, 6.0900, 7.9100	2.3600, 4.2700, 6.2700	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100	2.3600, 4.2700, 6.2700	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100	2.3600, 4.2700, 6.2700	5.1800, 7.1800, 8.9100
T51	2.4500, 4.2700, 6.2700	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100	2.0900, 3.9100, 5.8200	2.4500, 4.2700, 6.2700	2.0900, 3.9100, 5.8200	2.4500, 4.2700, 6.2700	2.0900, 3.9100, 5.8200	4.0900, 6.0900, 7.9100	2.0900, 3.9100, 5.8200
T52	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	3.0900, 5.0000, 6.8200	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	4.0900, 6.0900, 7.9100	3.0900, 5.0000, 6.8200	5.1800, 7.1800, 8.9100	3.0900, 5.0000, 6.8200

Table 5.15: The Normalized Fuzzy-Decision Matrix

	HWA1	**HWA2**	**HWA3**	**HWA4**	**HWA5**	**HWA6**	**HWA7**	**HWA8**	**HWA9**	**HWA10**
T11	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.5200, 0.7400, 0.9400	0.5200, 0.7400, 0.9400	0.5100, 0.7200, 0.9000
T12	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900
T13	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5900, 0.8000, 0.9700	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900
T21	0.4200, 0.6900, 0.9900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.1600, 0.4200, 0.7200	0.4200, 0.6900, 0.9900
T22	0.5200, 0.7400, 0.9400	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700
T23	0.6000, 0.8100, 1.0000	0.5200, 0.7400, 0.9400	0.5200, 0.7400, 0.9400	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400
T24	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.4300, 0.6400, 0.8600
T25	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700
T26	0.4200, 0.6900, 0.9900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400
T27	0.5900,	0.6000,	0.5000,	0.5900,	0.6000,	0.5000,	0.5900,	0.5200,	0.1600,	0.5700,

	0.8000, 0.9700	0.8100, 1.0000	0.7100, 0.8900	0.8000, 0.9700	0.8100, 1.0000	0.7100, 0.8900	0.8000, 0.9700	0.7400, 0.9400	0.4200, 0.7200	0.7800, 0.9600
T28	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700
T31	0.5200, 0.7400, 0.9400	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400
T41	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900
T42	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900
T43	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.4300, 0.6400, 0.8600
T44	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900
T51	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900	0.5900, 0.8000, 0.9700	0.6000, 0.8100, 1.0000	0.5000, 0.7100, 0.8900
T52	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900	0.5200, 0.7400, 0.9400	0.4200, 0.6900, 0.9900	0.5000, 0.7100, 0.8900

Table 5.16: The Weighted Normalized Fuzzy-Decision Matrix

	HWA1	**HWA2**	**HWA3**	**HWA4**	**HWA5**	**HWA6**	**HWA7**	**HWA8**	**HWA9**	**HWA10**
T11	0.00000, 0.00200, 0.00900	0.00200, 0.00900, 0.03000	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00200, 0.00700, 0.02400	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00200, 0.00700, 0.02400
T12	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04100	0.00200, 0.01000, 0.03700	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00900, 0.03000	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200
T13	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04200	0.00100, 0.00500, 0.01900	0.00300, 0.01100, 0.03600	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04100	0.00000, 0.00200, 0.00900	0.00200, 0.00900, 0.03000	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800
T21	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00500, 0.01600, 0.04800	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04200	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04100	0.00200, 0.01000, 0.03700	0.00100, 0.00500, 0.01800
T22	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00200, 0.00700, 0.02400	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04200	0.00100, 0.00500, 0.01900	0.00300, 0.01100, 0.03600
T23	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00500, 0.01600, 0.04800
T24	0.00000, 0.00200, 0.00900	0.00200, 0.00900, 0.03000	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00900, 0.03000	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00200, 0.00700, 0.02200	0.00200, 0.00700, 0.02400
T25	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04100	0.00200, 0.01000, 0.03700	0.00100, 0.00500, 0.01800	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04100	0.00200, 0.01000, 0.03700	0.00100, 0.00500, 0.01800	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800
T26	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04200	0.00100, 0.00500, 0.01900	0.00300, 0.01100, 0.03600	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04200	0.00100, 0.00500, 0.01900	0.00300, 0.01100, 0.03600	0.00200, 0.01000, 0.03700	0.00100, 0.00500, 0.01800

T27	0.00000, 0.00200, 0.00900	0.00200, 0.00900, 0.03000	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01900	0.00300, 0.01100, 0.03600
T28	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04100	0.00200, 0.01000, 0.03700	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00900, 0.03000	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900
T31	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04200	0.00100, 0.00500, 0.01900	0.00300, 0.01100, 0.03600	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04100	0.00200, 0.01000, 0.03700	0.00100, 0.00500, 0.01800	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900
T41	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00500, 0.01600, 0.04800	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04200	0.00000, 0.00200, 0.00900	0.00200, 0.00900, 0.03000	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800
T42	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00200, 0.00700, 0.02400	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04100	0.00200, 0.01000, 0.03700	0.00100, 0.00500, 0.01800
T43	0.00000, 0.00200, 0.00900	0.00200, 0.00900, 0.03000	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04200	0.00100, 0.00500, 0.01900	0.00300, 0.01100, 0.03600
T44	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04100	0.00200, 0.01000, 0.03700	0.00100, 0.00500, 0.01800	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00500, 0.01600, 0.04800
T51	0.00400, 0.01400, 0.04400	0.00300, 0.01200, 0.04200	0.00100, 0.00500, 0.01900	0.00300, 0.01100, 0.03600	0.00300, 0.01100, 0.03600	0.00400, 0.01400, 0.04400	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00200, 0.00700, 0.02400
T52	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00200, 0.00700, 0.02200	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800	0.00400, 0.01400, 0.04400	0.00100, 0.00500, 0.01800	0.00000, 0.00200, 0.00900	0.00500, 0.01600, 0.04800	0.00100, 0.00500, 0.01800

Table 5.17: Closeness Coefficients to the Aspired Level among the Different Alternatives

Alternatives (A)	di+	di−	Gap Degree of CCi+	Satisfaction Degree
HWA1	0.0454474	0.0268745	0.3667985	0.63224587
HWA2	0.0368854	0.0345687	0.4694754	0.52248579
HWA3	0.0364563	0.0421154	0.5838745	0.48518563
HWA4	0.0367745	0.0259665	0.4831957	0.57487563
HWA5	0.0401124	0.0468745	0.5348574	0.46799654
HWA6	0.0327741	0.0477456	0.6447526	0.35978546
HWA7	0.0452347	0.0265623	0.3961698	0.61385652
HWA8	0.0349578	0.0433748	0.5367895	0.46385634
HWA9	0.0390114	0.0468874	0.5335587	0.46678956
HWA10	0.0450448	0.0265574	0.3961965	0.61305689

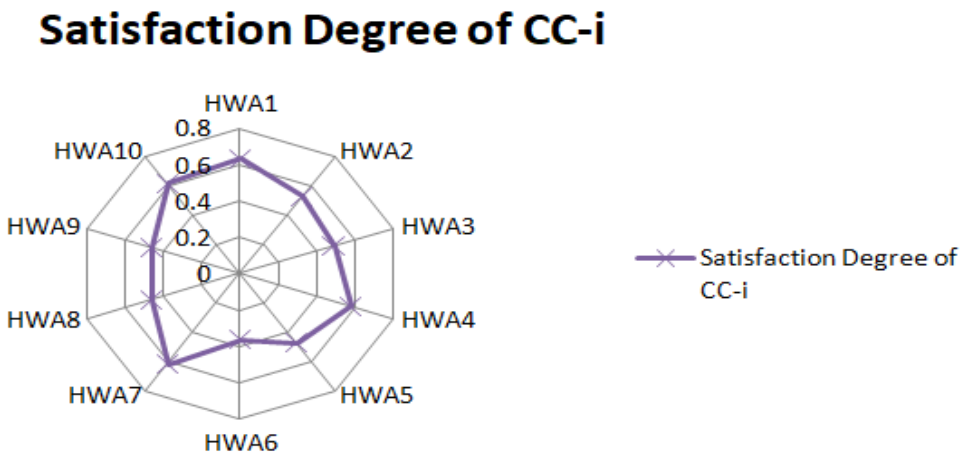

Figure 5.4: Satisfaction Degree of CC-i

Finally, the obtained global weight of factors from fuzzy AHP is sent to fuzzy TOPSIS approach as input, which generates a rank for alternatives. Now, the performance has been tested by using fuzzy AHP-TOPSIS. The determined performance of ten healthcare web application's alternatives is as: HWA1, HWA7, HWA10, HWA4, HWA2, HWA3, HWA5, HWA9, HWA8 and HWA6. According to the findings of our study, HWA1 produce the best result.

5.3.1 Sensitivity Analysis

Sensitivity analysis is threat to validity procedure that allows security practitioners to validate their results through numerical calculations. Additionally, threat to validity confers the idea to security experts on how various sources of outcomes may affect the proposed model. This section provides a clear understanding about the effectiveness as well as certainty of the results by altering the crucial criteria. For testing the sensitivity analysis, the researcher has chosen 10 alternatives in order to implement threat to validity. The detail of analyzed results of sensitivity analysis is shown in Table 5.18. Furthermore, a graphical representation of sensitivity analysis is depicted in Figure 5.5 for easy and detailed information.

Table 5.18: Sensitivity Analysis

	HWA1	HWA2	HWA3	HWA4	HWA5	HWA6	HWA7	HWA8	HWA9	HWA10
T0	0.63224587	0.52248579	0.48518563	0.57487563	0.46799654	0.35978546	0.61385652	0.46385634	0.46678956	0.61305689
T11	0.62554574	0.51758567	0.47428856	0.56949965	0.46159665	0.35938857	0.61116525	0.45529658	0.46159658	0.61119658
T12	0.64784584	0.52758547	0.48729658	0.58095568	0.47101122	0.35984459	0.61616324	0.47128564	0.47107854	0.61615564
T13	0.64426653	0.53305526	0.48027451	0.58695567	0.47552211	0.36138874	0.61911124	0.47925565	0.47554521	0.61917458
T21	0.63525564	0.71055264	0.49285526	0.59685564	0.46633322	0.37034456	0.61586321	0.45622541	0.46635224	0.61589654
T22	0.72137584	0.61415263	0.57088564	0.65545547	0.55436632	0.44286659	0.63811254	0.54872635	0.55432254	0.63811256
T23	0.67405564	0.56515564	0.52608547	0.61214485	0.50792266	0.39918547	0.65082231	0.49522635	0.50795263	0.65083251
T24	0.62955566	0.52728547	0.48105565	0.56255595	0.46615522	0.35414585	0.60818521	0.44325247	0.46612125	0.60811236
T25	0.62505564	0.53115563	0.47648521	0.54825568	0.46594444	0.34865265	0.60056354	0.43829635	0.46595263	0.60055214
T26	0.64498547	0.51558575	0.48322241	0.57695567	0.45927747	0.36922354	0.61414533	0.46425314	0.45921147	0.61412231
T27	0.64325654	0.52659965	0.48471122	0.57555569	0.46294415	0.36486325	0.61416325	0.46421125	0.46294456	0.61411133
T28	0.62728562	0.50258569	0.48623363	0.57355564	0.47021125	0.35522641	0.61262237	0.46225246	0.47027458	0.61262233
T31	0.61968547	0.49758574	0.48706636	0.57255574	0.47352263	0.35022357	0.61167596	0.46122563	0.47355474	0.61165566
T41	0.70728567	0.74804458	0.56472254	0.67124458	0.55452214	0.44223369	0.65867584	0.55472233	0.55457596	0.65862233
T42	0.66998564	0.55455596	0.52322265	0.62025569	0.50851125	0.40852235	0.65814569	0.50872211	0.50855235	0.65816611
T43	0.67428562	0.56155458	0.51378854	0.68595523	0.50502236	0.39984512	0.58218547	0.50421177	0.50502534	0.58217744
T44	0.72925623	0.60208854	0.55325564	0.65896622	0.54602234	0.44185263	0.69214596	0.54729965	0.54605234	0.69216655
T51	0.59085264	0.48215596	0.44822236	0.60652256	0.42911129	0.32382341	0.52255687	0.40505547	0.42911254	0.52258569
T52	0.59088547	0.48555565	0.44075564	0.53644447	0.42902254	0.32181125	0.57617858	0.42426654	0.42901254	0.57617456

The first row of table 5.18 shows the original weights of this study. The calculated results of table 5.18 are acceptable, and it is clear from above table that the deviation in the whole security risk factors is negligible. The results of sensitivity analysis are dependent on the weight of the security risk factors.

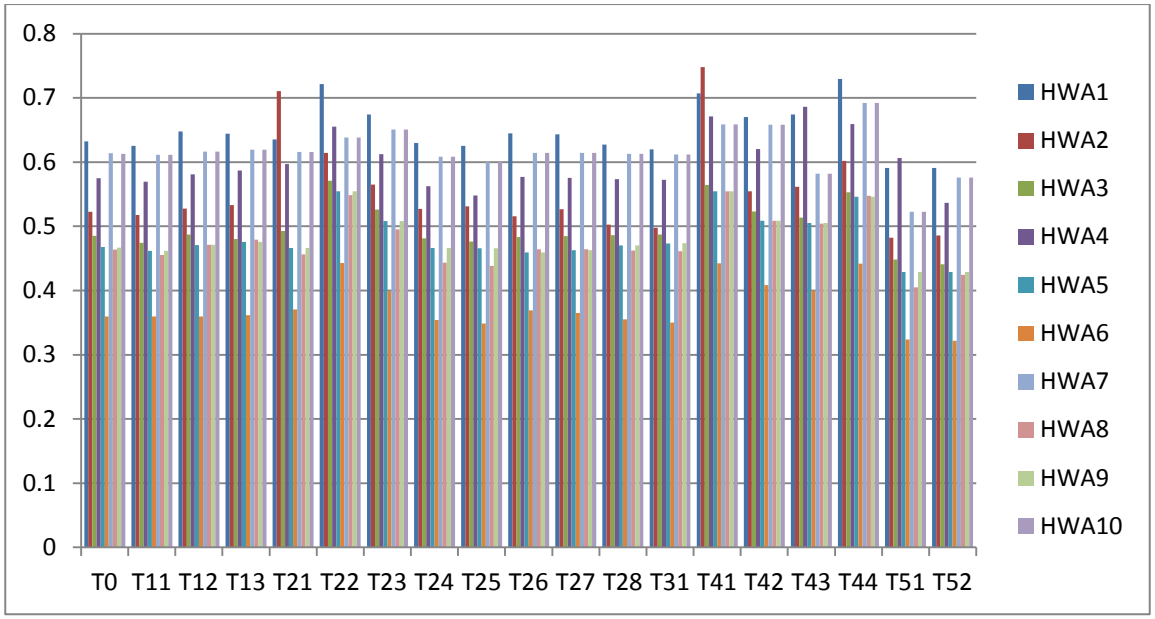

Figure 5.5: Graphical View of Sensitivity Analysis

Additionally, the researcher has used a confusion matrix for the assessment of the correctness of the outcomes and standard error rate. The confusion matrix is a scientific method. This is used for assessing the performance of the proposed method and its outcomes. Table 5.19 depicts a 2*2 dimensional confusion matrix. It works on the basis of 4 types of numbers, including TP, TN, FP, FN (True Positive, True Negative, False Positive, and False Negative respectively).

TABLE 5.19: Confusion Matrix

Actual/Predicted		Predicted	
		NO	**YES**
Actual	**NO**	[TN][1]	[FP][3]
	YES	[FN][2]	[TP][4]

Furthermore, in our case when the resources are changed and the values didn't vary then it is denoted by 1TN. The values that do not fluctuate as predicted by the researcher are denoted by 2FN. The values that fluctuate as predicted by researcher are denoted by 3FP, and the

values that fluctuate actually is denoted by 4TP. On the basis of table 5.19, the researcher evaluates the accuracy of results with the help of equation (27).

$$Accuracy = \frac{TP+TN}{TN+FP+FN+TP} \quad (27)$$

After evaluating the accuracy, the researcher has calculated the error rate by using equation (28).

$$Error\ Rate = \frac{FP+FN}{TN+FP+FN+TP} \quad (28)$$

The researcher has calculated the average accuracy of the result is 82%, and the error rate is 0.5% on the basis of equations (27) and (28) respectively. The results from the confusion matrix calculation show that the assessed results are extremely accurate and have a low error rate.

5.4 Comparison of the Results

MCDM approaches are used in a number of research initiatives to assess various factors and their impact on various fields. Comparison of results from different approaches may provide a considerable as well as clear perspective on computed results. In addition, comparing the outcomes of the same data through different approaches is a crucial part of scientific calculation.

For comparing the results of Fuzzy AHP-TOPSIS, the researcher has used various approaches, including Fuzzy Weighted Method, Fuzzy ANP-TOPSIS, Classical ANP-TOPSIS, Classical AHP-TOPSIS, and Simple Average Method.

Table 5.20: Comparison through Fuzzy AHP-TOPSIS Technique

Alternatives	Fuzzy AHP-TOPSIS	Fuzzy Weighted Method	Fuzzy ANP-TOPSIS	Classical AHP-TOPSIS	Classical ANP-TOPSIS	Simple Average Method
HWA1	0.63224587	0.63225632	0.63455655	0.62458577	0.62229966	0.62028658
HWA2	0.52248579	0.52255236	0.53595565	0.51694454	0.50355565	0.51305568
HWA3	0.48518563	0.48525523	0.49725547	0.47314466	0.46115544	0.47275565
HWA4	0.57487563	0.57495215	0.58164458	0.56855585	0.56185569	0.56398596
HWA5	0.46799654	0.46652263	0.46972265	0.45865574	0.45546658	0.45755566
HWA6	0.35978546	0.35982254	0.33232235	0.36834459	0.39585574	0.35835547
HWA7	0.61385652	0.61312236	0.59825562	0.61615535	0.63104459	0.60719965
HWA8	0.46385634	0.46321235	0.48865523	0.46162263	0.43625568	0.44825569
HWA9	0.46678956	0.46652215	0.46975564	0.45862257	0.45545567	0.45755535
HWA10	0.61305689	0.61311125	0.59822285	0.61617758	0.63105521	0.60712235

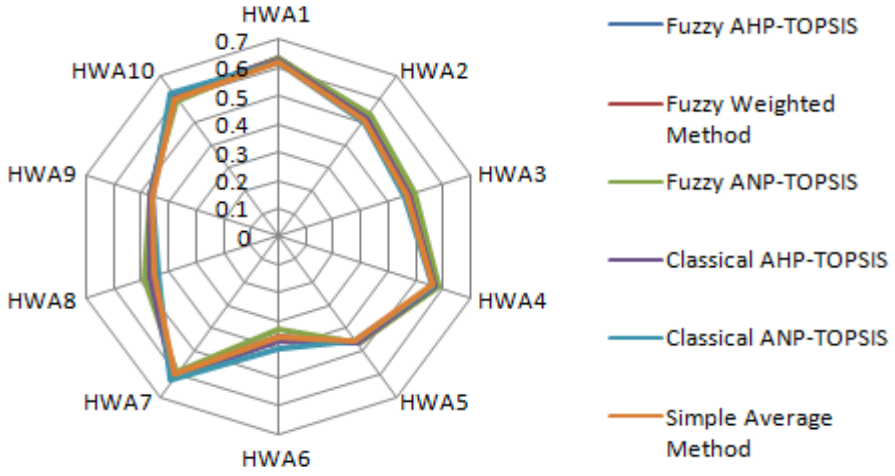

Figure 5.6: Comparison of Results

The capabilities and accuracy of the chosen approach are illustrated in this type of comparison. In comparison to the preceding techniques, the results of the fuzzy AHP-TOPSIS can confer a little more precise and preferable result, as shown in table 5.20.

5.5 Empirical Validation

Validity is basically the "measure of what is intended to be measured to the field", i.e. it explains how the recorded data is related to

the field under study. Validation is when results and specifications meet the requirements. This is the process to confirm that the developed application may fulfil the intended purpose. Validation is a method of deriving the truthfulness of any application. It acts as a benchmark to test the accuracy of the work done during development for which the requirements were collected and analysed. It can also be defined as when the software satisfies or meets the user requirements, or in other words validation is a process that leads to the acceptance of any model, approach, methodology, etc.

There are mainly two categories of validation processes, namely theoretical and empirical. Theoretical validation consigns to the estimation of the measure being taken for the quantum of attributes used in the development of the application. It addresses the question that whether the measurements being undertaken for the estimation of the attributes are true to user requirements. On the other hand, empirical validation refers to the question of measures taken or is useful in the impression that relates it to other variables in a typical way.

Table 5.21: Impact of Alternatives (Fuzzy AHP-TOPSIS Method Vs Fuzzy Weighted Method)

Alternatives/Methods	Fuzzy AHP-TOPSIS	Fuzzy Weighted Method
HWA1	0.63224587	0.63225632
HWA2	0.52248579	0.52255236
HWA3	0.48518563	0.48525523
HWA4	0.57487563	0.57495215
HWA5	0.46799654	0.46652263
HWA6	0.35978546	0.35982254
HWA7	0.61385652	0.61312236
HWA8	0.46385634	0.46321235
HWA9	0.46678956	0.46652215
HWA10	0.61305689	0.61311125

Table 5.21shows the impact of alternatives by comparing two distinct methods (Fuzzy AHP-TOPSIS Method Vs Fuzzy Weighted Method) on all the selected parameters. This shows the weight metrics of HWA.

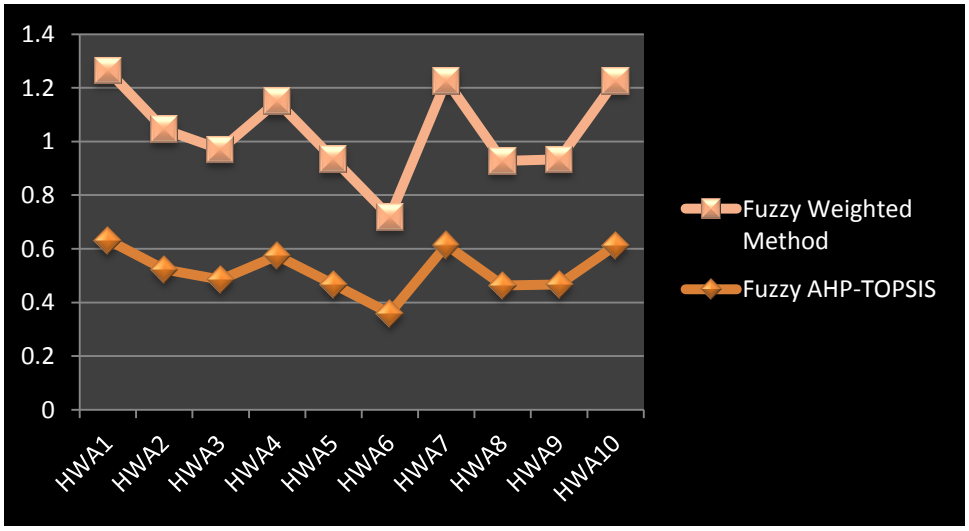

Figure 5.7: Graphical Representation of Comparison between Fuzzy AHP-TOPSIS Method and Fuzzy Weighted Method

The comparative study of above table 5.21 and results has been presented in figure 5.7. Hence it is found the results obtained from both the methods are close and bound to each other.

Table 5.22: Impact of Alternatives (Fuzzy AHP-TOPSIS Method Vs Fuzzy ANP-TOPSIS Method)

Alternatives/Methods	Fuzzy AHP-TOPSIS	Fuzzy ANP-TOPSIS
HWA1	0.63224587	0.63455655
HWA2	0.52248579	0.53595565
HWA3	0.48518563	0.49725547
HWA4	0.57487563	0.58164458
HWA5	0.46799654	0.46972265
HWA6	0.35978546	0.33232235
HWA7	0.61385652	0.59825562
HWA8	0.46385634	0.48865523
HWA9	0.46678956	0.46975564
HWA10	0.61305689	0.59822285

Table 5.22 shows the impact of alternatives by comparing two distinct methods (Fuzzy AHP-TOPSIS Method Vs Fuzzy ANP-TOPSIS Method) on all the selected parameters. This shows the weight metrics of HWA.

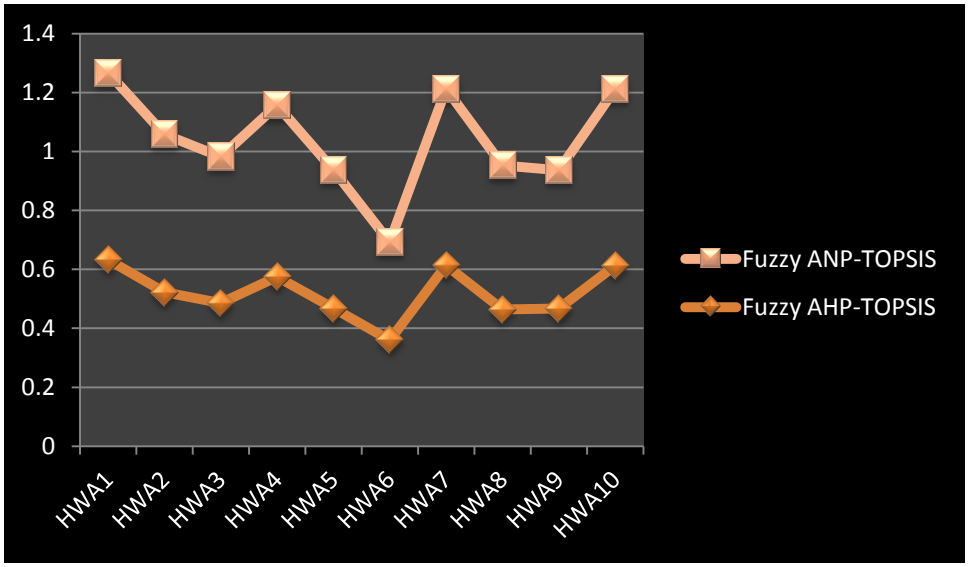

Figure 5.8: Graphical Representation of Comparison between Fuzzy AHP-TOPSIS Method and Fuzzy ANP-TOPSIS Method

The comparative study of above table 5.22 and results has been presented in figure 5.8. Hence it is found the results obtained from both the methods are close and bound to each other.

Table 5.23: Impact of Alternatives (Fuzzy AHP-TOPSIS Method Vs Classical AHP-TOPSIS Method)

Alternatives/Methods	Fuzzy AHP-TOPSIS	Classical AHP-TOPSIS
HWA1	0.63224587	0.62458577
HWA2	0.52248579	0.51694454
HWA3	0.48518563	0.47314466
HWA4	0.57487563	0.56855585
HWA5	0.46799654	0.45865574
HWA6	0.35978546	0.36834459
HWA7	0.61385652	0.61615535
HWA8	0.46385634	0.46162263
HWA9	0.46678956	0.45862257
HWA10	0.61305689	0.61617758

Table 5.23 shows the impact of alternatives by comparing two distinct methods (Fuzzy AHP-TOPSIS and Classical AHP-TOPSIS) on all the selected parameters. This shows the weight metrics of HWA.

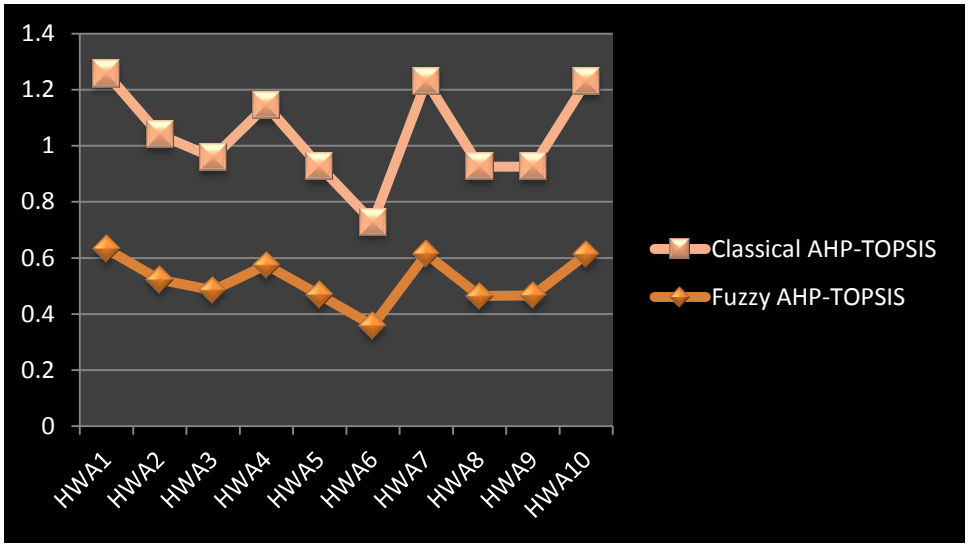

Figure 5.9: Graphical Representation of Comparison between Fuzzy AHP-TOPSIS Method and Classical AHP-TOPSIS Method

The comparative study of above table 5.23 and results has been presented in figure 5.9. Hence it is found the results obtained from both the methods are close and bound to each other.

Table 5.24: Impact of Alternatives (Fuzzy AHP-TOPSIS Method Vs Classical ANP-TOPSIS Method)

Alternatives/Methods	Fuzzy AHP-TOPSIS	Classical ANP-TOPSIS
HWA1	0.63224587	0.62229966
HWA2	0.52248579	0.50355565
HWA3	0.48518563	0.46115544
HWA4	0.57487563	0.56185569
HWA5	0.46799654	0.45546658
HWA6	0.35978546	0.39585574
HWA7	0.61385652	0.63104459
HWA8	0.46385634	0.43625568
HWA9	0.46678956	0.45545567
HWA10	0.61305689	0.63105521

Table 5.24 shows the impact of alternatives by comparing two distinct methods (Fuzzy AHP-TOPSIS and Classical ANP-TOPSIS) on all the selected parameters. This shows the weight metrics of HWA.

Figure 5.10: Graphical Representation of Comparison between Fuzzy AHP-TOPSIS Method and Classical ANP-TOPSIS Method

The comparative study of above table 5.24 and results has been presented in figure 5.10. Hence it is found the results obtained from both the methods are close and bound to each other.

Table 5.25: Impact of Alternatives (Fuzzy AHP-TOPSIS Method Vs Simple Average Method)

Alternatives/Methods	Fuzzy AHP-TOPSIS	Simple Average Method
HWA1	0.63224587	0.62028658
HWA2	0.52248579	0.51305568
HWA3	0.48518563	0.47275565
HWA4	0.57487563	0.56398596
HWA5	0.46799654	0.45755566
HWA6	0.35978546	0.35835547
HWA7	0.61385652	0.60719965
HWA8	0.46385634	0.44825569
HWA9	0.46678956	0.45755535
HWA10	0.61305689	0.60712235

Table 5.25 shows the impact of alternatives by comparing two distinct methods (Fuzzy AHP-TOPSIS and Simple Average Method) on all the selected parameters. This shows the weight metrics of HWA.

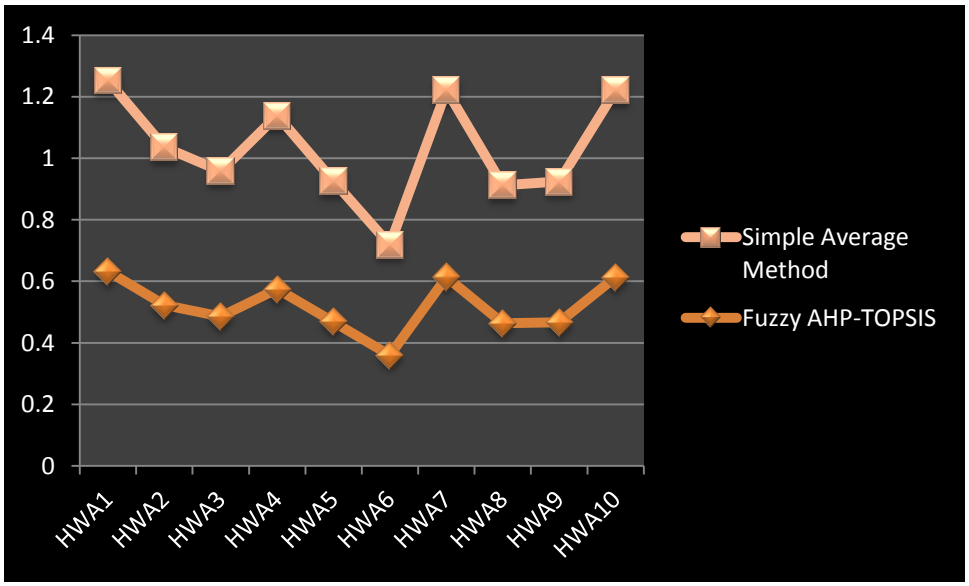

Figure 5.11: Graphical Representation of Comparison between Fuzzy AHP-TOPSIS Method and Simple Average Method

The comparative study of above table 5.25 and results has been presented in figure 5.11. Hence it is found the results obtained from both the methods are close and bound to each other.

Comparison of results with different methodologies may bestow significant as well as understandable results. Furthermore, the comparison of results of the same data through various approaches is an important part of scientific calculation.

5.6 Statistical Analysis

Statistics is the branch of mathematics that deals with gathering, organizing, analysing, interpreting and presenting data. The process of collection and interpretation of information in order to identify the trends and patterns is known as statistical analysis. The F-test is used to compare statistics-based models that have been fitted to a dataset. F-test usually

comes into play when a model is fitted to the data through the least-squares method.

5.6.1 Hypothesis Testing

Hypothesis testing is the statistical analysis used to take decisions based on statistics of experimental data. An analysis that is statistical in nature is the basis of the assumption of hypothesis testing. The null hypothesis shows that there is no significant relationship among two or more parameters, while the alternate hypothesis confirms the relationship among the concerned parameters. In other words, acceptance of an alternate hypothesis depends upon the affirmation of the relationship between parameters or rejection of the null hypothesis.

Table 5.26: Results between Fuzzy AHP-TOPSIS Method and Fuzzy Weighted Method

F-Test Two-Sample for Variances		
	Variable 1	**Variable 2**
Sample Average (Mean)	0.520013	0.519733
Sample Standard Deviation	0.0873396	0.0874179
Observations	10	10
Degree of Freedom	9	9
F	0.9979	
F-Critical one-tail	0.9982	

Null hypothesis (H0): There is no significant difference between Fuzzy AHP-TOPSIS Method and Fuzzy Weighted Method.

H0: $\mu 1 - \mu 2 = 0$

Alternate hypothesis (Ha): There is a significant difference between Fuzzy AHP-TOPSIS Method and Fuzzy Weighted Method.

Ha: $\mu 1 - \mu 2 \neq 0$

Table 5.26 shows the final outcomes of Fuzzy AHP-TOPSIS Method and Fuzzy Weighted Method, and difference between the results of both methods are negligible. Further, the calculated F-value is 0.9979, which is less than the value of F-critical value i.e., 3.179 for one tail test at the 0.05 level for 9 degree of freedom. Hence the null hypothesis is accepted, and the alternate hypothesis rejected.

Table 5.27: Results between Fuzzy AHP-TOPSIS Method and Fuzzy ANP-TOPSIS Method

F-Test Two-Sample for Variances		
	Variable 1	**Variable 2**
Sample Average (Mean)	0.520013	0.520635
Sample Standard Deviation	0.0873396	0.0889392
Observations	10	10
Degree of Freedom	9	9
F	0.9644	
F-Critical one-tail	0.9578	

Null hypothesis (H0): There is no significant difference between Fuzzy AHP-TOPSIS Method and Fuzzy ANP-TOPSIS Method.

H0: $\mu 1 - \mu 2 = 0$

Alternate hypothesis (Ha): There is a significant difference between Fuzzy AHP-TOPSIS Method and Fuzzy ANP-TOPSIS Method.

Ha: $\mu 1 - \mu 2 \neq 0$

Table 5.27 shows the final outcomes of Fuzzy AHP-TOPSIS Method and Fuzzy ANP-TOPSIS Method, and difference between the results of both methods are negligible. Further, the calculated F-value is 0.9644, which is less than the value of F-critical value i.e., 3.179 for one tail test at the 0.05 level for 9 degree of freedom. Hence the null hypothesis is accepted, and the alternate hypothesis rejected.

Table 5.28: Results between Fuzzy AHP-TOPSIS Method and Classical AHP-TOPSIS Method

F-Test Two-Sample for Variances		
	Variable 1	**Variable 2**
Sample Average (Mean)	0.520013	0.516281
Sample Standard Deviation	0.0873396	0.0867921
Observations	10	10
Degree of Freedom	9	9
F	0.9854	
F-Critical one-tail	1.0127	

Null hypothesis (H0): There is no significant difference between

H0: $\mu 1 - \mu 2 = 0$

Alternate hypothesis (Ha): There is a significant difference between Fuzzy AHP-TOPSIS Method and Classical AHP-TOPSIS Method.

Ha: $\mu 1 - \mu 2 \neq 0$

Table 5.28 shows the final outcomes of Fuzzy AHP-TOPSIS Method and Classical AHP-TOPSIS Method, and difference between the results of both methods are negligible. Further, the calculated F-value is 0.9854, which is less than the value of F-critical value i.e., 3.179 for one tail test at the 0.05 level for 9 degree of freedom. Hence the null hypothesis is accepted, and the alternate hypothesis rejected.

Table 5.29: Results between Fuzzy AHP-TOPSIS Method and Classical ANP-TOPSIS Method

F-Test Two-Sample for Variances		
	Variable 1	**Variable 2**
Sample Average (Mean)	0.520013	0.5154
Sample Standard Deviation	0.0873396	0.0889183
Observations	10	10
Degree of Freedom	9	9
F	0.9583	
F-Critical one-tail	0.9648	

Null hypothesis (H0): There is no significant difference between Fuzzy AHP-TOPSIS Method and Classical ANP-TOPSIS Method.

H0: $\mu 1-\mu 2 = 0$

Alternate hypothesis (Ha): There is a significant difference between Fuzzy AHP-TOPSIS Method and Classical ANP-TOPSIS Method.

Ha: $\mu 1-\mu 2 \neq 0$

Table 5.29 shows the final outcomes of Fuzzy AHP-TOPSIS Method and Classical ANP-TOPSIS Method, and difference between the results of both methods are negligible. Further, the calculated F-value is 0.9583, which is less than the value of F-critical value i.e., 3.179 for one tail test at the 0.05 level for 9 degree of freedom. Hence the null hypothesis is accepted, and the alternate hypothesis rejected.

Table 5.30: Results between Fuzzy AHP-TOPSIS Method and Simple Average Method

F-Test Two-Sample for Variances		
	Variable 1	**Variable 2**
Sample Average (Mean)	0.520013	0.510613
Sample Standard Deviation	0.0873396	0.0867052
Observations	10	10
Degree of Freedom	9	9
F	0.983	
F-Critical one-tail	1.0147	

Null hypothesis (H0): There is no significant difference between Fuzzy AHP-TOPSIS Method and Simple Average Method.

H0: $\mu 1-\mu 2 = 0$

Alternate hypothesis (Ha): There is a significant difference between Fuzzy AHP-TOPSIS Method and Simple Average Method.

Ha: $\mu 1-\mu 2 \neq 0$

Table 5.30 shows the final outcomes of Fuzzy AHP-TOPSIS Method and Simple Average Method, and difference between the results of both methods are negligible. Further, the calculated F-value is 0.983, which is less than the value of F-critical value i.e., 3.179 for one tail test at the 0.05 level for 9 degree of freedom. Hence the null hypothesis is accepted, and the alternate hypothesis rejected.

The F-test analysis applies for verifying the significance between the Fuzzy AHP-TOPSIS approach and other traditional approaches, which are shown in Table 5.26 to 5.30. The table 5.26 to 5.30 shows the variance value of two variables means that squared differences from the mean of fuzzy AHP-TOPSIS and other traditional approaches simultaneously. On the basis of hypothesis test, it is observed that the complete value of our research hypothesis and easy to choose the kind of best-suited test for research work. In this study, six independent approaches for estimating the impact of security risk factors are used on healthcare web applications. We shall test the hypothesis at 95% confidence interval. Since there are 10 observations, therefore the degree of freedom is 9. Hence, it is validated that the risk assessment rate of the Fuzzy AHP-TOPSIS approach is better than those of traditional approaches.

5.7 Conclusion

Security has a significant impact on the overall healthcare web application and it significantly influences the Healthcare Information System's trustworthiness. To address security risk factors and to evaluate the impact of risk factors on the hospital information system, the current endeavour is assessed through MCDM techniques. The researcher has used an integrated fuzzy AHP-TOPSIS approach for assessment because these are highly effective techniques for addressing decision-making problems and provide efficient results. On the basis of expert's opinions and current research findings, the identification and selection of security risk factors

for evaluation have been made. Further, this assessment is validated by F-test. This validation establishes the fact that the proposed work has given the framework that is satisfactory from a security perspective. Hence, the null hypothesis (H0) formulated at the beginning of statistical analysis is accepted, and the alternative hypothesis (Ha) is rejected. Therefore, the researcher claims that the integrated approach, i.e., the fuzzy AHP-TOPSIS approach provides better assessment and gives a better result as compare to traditional approaches.

Chapter 6
SUMMARY AND CONCLUSIONS

6.1 Introduction

This chapter summarizes the main contributions of the research and also presents future work in the area of creating a secure healthcare web application. The early detection and estimation of security issues are beneficial for the healthcare web application. This framework provides a blueprint and a comprehensive roadmap for making viable transformations in designing a secure application. A perspective security framework for assessing security through security attributes revealed many things that include good security models and methods to assess security risk in an application from start-to-end. The current study involves various aspects for designing a security risk estimation framework, such as identification of factors, mapping of security risk factors with their corresponding security attributes, assessment of risk factors, statistical analysis and review and revision. In addition, an integrated fuzzy AHP-TOPSIS with MCDM is also included in this research. The integrated fuzzy AHP-TOPSIS approach effectively analyses any MCDM issue with various alternatives and variables.

In this research, a comprehensive review of literature has made for identifying the pertinent security risk at early stage of development and then a hierarchical structure of attributes was proposed. Next, the opinions of various experts are collected. These experts are from the software field and academia. The main objective of this study is to estimate the impact of security attributes on the healthcare web application. Various attributes of security that has an impact on healthcare web application is evaluated as well as estimated. Furthermore, their weights are calculated, and the ranking of the alternatives are

determined. The results obtained from this study would assist the security practitioners to identify and prioritize the most effective risk factor.

6.2 Significant Contributions

A depth study of the existing literature on healthcare information systems and a challenging research process done by the researcher has made the following major contributions:

- **Some Common Software Security Risk Factors at Design Phase**

  A significant review has been done on some of the most common security risk factors on the basis of a literature survey during the preliminary stages of research work. It has been perceived that from the past few years, research in the field of security has been going on worldwide. Software security is currently one of the most important concerns to the world as researchers depend on information technology rapidly. Continuous growth and heavy dependency on information technology and IT-based resources have produced a new problem because attackers have adopted new attack mechanisms to break the software's confidentiality, integrity, and authenticity. So, security is the elementary requirement of secure software development. The development of secure and software products is the demand of time. Identification and prioritization of security risk factors is the first step towards providing a secure environment.

- **A Pertinent Review on Some Software Security Risk Management Frameworks**

  After reviewing the several available literatures in the area of security risk management framework, the researcher has done a detailed review on some pertinent security risk management frameworks. At last, the researcher has concluded that there is need of a framework to reduce risk, with most effective controls & uses a low-cost strategy, and should be tested on a large scale.

- **Major Software Security Risks and their Mitigation Strategies: A Design Phase Perspective**

  In this contribution researcher have suggested mitigation strategy for the previously identified security risk factors, i.e., published in "Some Common Software Security Risk Factors at Design Phase".

- **Estimation of Software Security Risks through CVSS: A Design Phase Perspective**

  In this contribution, the researcher has laid stress on software security risk at the design phase. They have listed most common security risk and from CWE (i.e., Common Weakness Enumeration). Furthermore, the researcher used the CVSS 3.1 mechanism to calculate the scores of identified risks. The objective of the study is to prioritize the impact of identified security risk factors.

6.2.1 Other Findings

- A Phase-wise Review of Software Security Metrics
- An Enhancement of Two-Tier ATM Security Framework
- Prediction of COVID-19 Pandemic Spread in Kingdom of Saudi Arabia
- A Critical Analysis of Fraud Cases on the Internet

6.3 Research Findings

During the course's study, the research objective has to find out the answers to research questions posed in 1st chapter. In this, the researcher has also tried to solve the security challenges acknowledged during the survey of the literature. The present section answers the research questions raised in 1st chapter.

- What are the factors that directly influence the security of healthcare web applications?

Research Finding: Confidentiality, Integrity, Availability, Access Control and Authentication are the factors that directly influence the security of healthcare web applications.

- Is there any standard framework available for estimating security risk through the MCDM technique for healthcare web application?

  Research Finding: The literature survey reveals that there is no standard framework is available for estimation of security risk through the MCDM technique for the healthcare web application. The researcher made a contribution in this regard to develop and validate a perspective framework that assesses the security risk factors at the design phase of the healthcare web application.

- What are the major challenges with respect to secure healthcare web application development?

  Research Finding: A major challenge towards a secure healthcare web application development is the lack of security knowledge and expertise among ordinary security developers.

- Can we develop an integrated security approach that incorporates all the security risk factors?

  Research Finding: Yes, we can develop an integrated approach that incorporates all security factors.

- Can we develop a framework that may be used in the design phase to estimate the security risk of the healthcare web application?

  Research Finding: yes, it is feasible to develop a framework that may be used in the design phase to estimate the security risk of the healthcare web application.

- Which security risk attributes needs to be focused upon according to their respective weightage?

  Research Finding: T31 attribute needs to be focused upon according to their respective weightage.

- How general are the lessons learned in this study? Can they be applied in situations involving other environments or organizations with different operational contexts?

  Research Finding: The current study has proposed a security risk estimation framework for the healthcare web application, using which various models and mechanisms are developed.

6.4 Future Directions

Research is an ongoing activity. Reaching one milestone encourages the way to the next. As a future research plan, there may be the following tasks to be performed:

- In future, the researcher has intended to implement the proposed framework for other big healthcare projects.
- The proposed framework for the estimation of the security risk of the healthcare web application can be modified so that it can help to maximize the security of all types of healthcare web applications not limited to given alternatives.
- The researcher will develop an automatic tool for the proposed framework to elicit security risk attributes more efficiently.
- The researcher is planning to conduct more experiments with the help of different healthcare web applications to draw more concrete conclusions.
- The researcher also plans to extend the proposed methodology with detailed features appropriate to several modern technologies, including Internet of Things (IoT), edge computing, cloud computing etc.
- Various implementations of the proposed framework for security risk assessment are possible. A researcher may implement the same framework by choosing another set of security risk attributes.

6.5 Conclusion

The main objective of this research is to assess security risk in healthcare web applications at the design phase. In addition, a comprehensive literature review has been made for the identification of security risk factors and security attributes affecting the secure development of healthcare web applications. After that, a hierarchical structure is fabricated. Furthermore, the opinion of various experts is collected. These experts are from the software field and academia.

In this research, for the estimation of security risk, an integrated fuzzy AHP-TOPSIS approach is applied. The weight and priority of risk factors are calculated with the help of fuzzy AHP, whereas the impact of attributes on different alternatives is calculated with the help of fuzzy AHP-TOPSIS. Furthermore, the comparison of results has been done and founded that the obtained weight is more accurate as compared to other techniques. In addition, the researcher suggested that the proposed framework may be used to set the benchmark for any organization. It may also form the basis of development of new, modified or refined approaches. The proposed framework may encourage other researchers to undertake the development of other new methods in this area.

ABSTRACT

The modern world is critically reliant on a broad range of software applications. Dependency on software applications is so high that life cannot be imagined without them. Information, no matter to which part of the globe it belongs, is available with a click of the mouse. Intensive security-oriented services ranging from internet banking, trading to online, buying and selling, booking an appointment to a doctor etc., are carried out unhesitatingly. These services require the privacy of the information and asset. When security intensive information is floating everywhere, anyone having malicious intent can misuse the information. This may harm an organization or individuals. Since decades, efforts are being made to estimate security risk in order to increase accountability, demonstrate compliance, and determine whether and by how much our investments in the product make our systems more secure.

Furthermore, the health sector is one of the most prime sectors where all the hi-tech applications are used. In this sector, medical personnel are entrusted with a vast number of responsibilities, and dealing with them is a more sophisticated as well challenging task. The healthcare sector has been linked to the technological world in order to ease the responsibilities as well as workloads of the healthcare staff. This was made possible by integrating IT (Information Technology) into the healthcare sector. Apart from these technological advancements, several statistics have demonstrated data breaches instances that have affected both, i.e., patients and Healthcare Information Systems (HISs). Thousands of healthcare records can be compromised by security breaches.

In addition, to secure an individual's as well as HIS's data, three major security factors and privacy goals are needed, which is commonly known as the CIA triad. The significant necessity of the CIA trio is;

confidentiality must be included for highly sensitive data, integrity is important because it may be fatal to provide an inaccurate procedure based on faulty data of medical, and availability is necessary because the data must be available on time for adequate treatment.

In the healthcare web application, the privacy of individual and organizational data is extremely important, and currently it has become a major challenge to shield healthcare information. The major challenge introduced in the healthcare web application is due to huge data growth. Furthermore, COVID-19 (i.e., the current pandemic situation) has resulted in an unexpected spike in healthcare data, which has impacted both the healthcare web applications and hospitals. Managing these healthcare data and securing it from intruders has become a complex task for security experts. Nowadays, the main objective of the researchers and security experts is to minimize security vulnerabilities in the healthcare web application by mitigating and assessing the security risk factors. So, some dedicated steps are required to enhance the security of healthcare web applications, which may help in securing and protecting them in order to ensure transparency and assess security risk. This is why security professionals prefer to take a step up on the design phase to reduce security risks. It will assist in designing secure web applications in the healthcare sector. Furthermore, it may also assist in overcoming from threats and protecting it from cyber-attacks by early detection and mitigation of security risk factors in the design phase.

In order to gain a competitive edge, developers and researchers need to create a viable security risk assessment framework so as to minimize critical healthcare web application failures. Though it is highly difficult to create a perfectly secure healthcare web application system, but one can surely reduce the security risks by following a fool-proof and meticulously designed strategy with the inclusion of security attributes. In

addition to this, the researcher has made an effort to overcome this issue and proposed a framework to assess security risk of the healthcare web application. This framework incorporates five phases, including Factors Identification, Mapping, Assessment, Statistical Analysis and Review and Revision.

The first phase, i.e., Factors Identification, in which the identification and selection of security risk factors as well as their corresponding security attributes have been made on the basis of a comprehensive literature review and expert's opinions. The relationship among security attributes and security risk factors has been developed in the second phase. In addition, an integrated Fuzzy AHP-TOPSIS approach is used for security risk assessment in the third phase. Where Fuzzy AHP is used to prioritize security risk factors, and the impact of security attributes on various alternatives is calculated with the help of Fuzzy AHP-TOPSIS. Furthermore, sensitivity analysis and empirical validation are carried out in the second last phase of the framework. In the last phase, review and revision will be undertaken only when required, which facilitates a retrospect of the entire development activity and aid in making changes whenever necessary.

It is apparent from the validation of the proposed framework that it may be significantly helpful to keep in check the potential risk and vulnerabilities from the early design phase till the end. The proposed framework has shown satisfactory results with respect to other mentioned approaches. It may also form the basis for the development of new modified or refined approaches. Like any other research, the current work may also suffer from certain limitations, therefore to achieve a generalized result and implementation of the proposed model, further study may be conducted on large applications.

B. Main References

1. Jesdabodi, C., & Maalej, W. (2015, September). Understanding Usage States on Mobile Devices. In Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing. New York, USA, pp: 1221-1225.
2. Savola, R. M., (2009). A Security Metrics Development Method for Software Intensive Systems. Advances in Information Security and its Application, Communications in Computer and Information Science, 2009, Springer, Vol. 36, pp: 11-16.
3. G. McGraw, Software Security, IEEE Security and Privacy, 2004, Vol. 2, No. 2, pp: 80-83.
4. McGraw, G. (2002). Managing Software Security Risks. Computer, Vol. 35, No. 4, pp: 99-101.
5. Available at: https://cwe.mitre.org/top25/archive/2020/2020_cwe_top25.html.
6. Lim, DE., & Kim, TS., (2014). Modelling Discovery and Removal of Security Vulnerabilities in Software System Using Priority Queuing Models. Journal of Computer Virology and Hacking Techniques, Springer, pp: 109–114.
7. Zarour, M., Alenezi, M., & Alsarayrah, K. (2020). Software Security Specifications and Design: How Software Engineers and Practitioners Are Mixing Things up. In Proceedings of the Evaluation and Assessment in Software Engineering pp: 451-456.
8. Sultan, K., En-Nouaary, A., & H-Lhadj, A., (2008). Catalog for Assessing Risks of Software Throughout the Software Development Life Cycle. In the Proc. of International Conference on Information Security and Assurance, IEEE. pp: 461-465.
9. Alenezi, M., & Almuairfi, S. (2019). Security risks in the software development lifecycle. International Journal of Recent Technology and Engineering, Vol. 8, No. 3, pp: 7048-7055.

10. McGraw, G. (2016). Four Software Security Findings. Computer, Vol. 49, No.1, pp: 84-87.
11. Abunadi, I., & Alenezi, M. (2016). An Empirical Investigation of Security Vulnerabilities within Web Applications. J. Univers. Comput. Sci., Vol. 22, No. 4, pp: 537-551.
12. Tiwari, Neeraj, Kumar Rai, Sahani Ashok, Anurag & Maurya, Akshay, (2019), Survey Paper on Hospital Management System (HMS), International Journal of Scientific Research and Review ISSN No.: 2279-543X, Vol. 07, No. 03, March 2019, UGC Journal No.: 64650
13. Available Online at: https://medium.com/@KNOWARTH/6-benefits-of-implementing-a-hospital-management-system-9cde28b5926a
14. Kaur, J., Khan, A. I., Abushark, Y. B., Alam, M. M., Khan, S. A., Agrawal, A. Kumar, Rajeev & Khan, R. A. (2020). Security Risk Assessment of Healthcare Web Application through Adaptive Neuro-fuzzy Inference System: A Design Perspective, Risk Management and Healthcare Policy, Vol. 13, pp: 355-371.
15. Chopra, M. (2013). IT Security in Hospital Management. Global Journal of Computer Science and Technology. Vol. 13 No. 3 Version 1.0, Publisher: Global Journals Inc. (USA), Online ISSN: 0975-4172 & Print ISSN: 0975-4350
16. Westin, A. F. (1976). Computers, health Records, and Citizen Rights (No. 157-158). US Department of Commerce, National Bureau of Standards.
17. Haas, S., Wohlgemuth, S., Echizen, I., Sonehara, N., & Müller, G. (2011). Aspects of Privacy for Electronic Health Records. International Journal of Medical Informatics, Vol. 80, No. 2, pp: 26-31.
18. Farzandipour, M., Sadoughi, F., Ahmadi, M., & Karimi, I. (2010). Security Requirements and Solutions in Electronic Health records:

Lessons Learned from a Comparative Study. Journal of medical systems, Vol. 34, No. 4, pp: 629-642.

19. Rindfleisch, T. C. (1997). Privacy, Information Technology, and Healthcare. Communications of the ACM, Vol. 40, No. 8, pp: 92-100.
20. Olivier, M. S. (2002). Database Privacy: Balancing Confidentiality, Integrity and availability. ACM SIGKDD Explorations Newsletter, Vol. 4, No. 2, pp: 20-27.
21. Joh, H., & Malaiya, Y. K. (2010, November). A Framework for Software Security Risk Evaluation Using the Vulnerability Lifecycle and Cvss Metrics. In Proc. International Workshop on Risk and Trust in Extended Enterprises, pp. 430-434.
22. Fernández-Alemán, J. L., Señor, I. C., Lozoya, P. Á. O., & Toval, A. (2013). Security and Privacy in Electronic Health Records: A systematic Literature Review. Journal of Biomedical Informatics, Vol. 46, No. 3, pp: 541-562.
23. Kim, C. Y., Lee, J. S., & Kim, Y. I. (2002). Early Stage Evolution of a Hospital Information System in a Middle Income Country: A Case study of Korea, International Journal of Healthcare Technology and Management, Vol. 4, No. 6, pp: 514-524.
24. World Health Organization. (1957). Role of Hospitals in Programmes of Community Health Protection: First Report of the Expert Committee on Organization of Medical Care [meeting held in Geneva from 18 to 23 June 1956]. World Health Organization. Series No. 122.
25. Chitkara, Mansi, Khandelwal, Namita & Chaporkar, Avinash (2010), Project Report on Hospital Management System. International School of Informatics & Management (Formerly India International Institute of Management)
26. Charles, Waban. (2007) Project Report on Computerized Health Records Management System.

27. Kumar, Prem & Kosalram, Kalpana. (2013). E–Hospital Management & Hospital Information Systems–Changing Trends. International Journal of Information Engineering and Electronic Business. pp: 50-58.
28. Available online at: https://www.macrofocusng.com/Rex-HMS /index.php
29. Available online at: https://www.fiverr.com/codexcube/provide-hospital-management-system
30. NHS Lothian Communications Office. NHS Lothian Staff Member Loses Patient Data. http://www.nhslothian.scot.nhs.uk /Media Centre/PressReleases/2008/Pages/0307PatientData.aspx/.
31. Department of Veterans Affairs Office of Inspector General. Review of Issues Related to the Loss of VA Information Involving the Identity of Millions of Veterans; July 11 2006, Report No.06-02238-163, Available Online at: https://www.va.gov/oig /pubs/ VAOIG-06-02238-163.pdf
32. Available online at: https://hitinfrastructure.com/news /healthcare-application-management-growth-seeks-containers#:~: text=Healthcare%20application%20management%20is%20a, available%20on%20their%20preferred%20devices.
33. Rothstein, M. A., & Talbott, M. K. (2007). Compelled Authorizations for Disclosure of Health Records: Magnitude and Implications. The American Journal of Bioethics, Vol. 7, No. 3, pp: 38-45.
34. Al Ameen, M., Liu, J., & Kwak, K. (2012). Security and Privacy Issues in Wireless Sensor Networks for Healthcare Applications. Journal of Medical Systems, Vol. 36, No. 1, pp: 93-101.
35. Available online at: https://healthitsecurity.com/news/ understanding-web-application-security-in-healthcare
36. Deng, Y., Wang, J., & Tsai, J. J. (2001, March). Formal Analysis of Software Security System Architectures. In Proceedings 5th

International Symposium on Autonomous Decentralized Systems, IEEE, pp: 426-434.

37. Walton, G. H., Longstaff, T. A., & Linger, R. C. (2006). Technology Foundations for Computational Evaluation of Software Security Attributes. Carnegie-Mellon Univ. Pittsburgh Pa Software Engineering Inst.
38. Available online at: https://www.betterhealth.vic.gov.au/health /ServicesAndSupport/confidentiality-and-privacy-in-healthcare
39. Khan, S. A., & Khan, R. A. (2012). Confidentiality Quantification Model at Design Phase. International Journal of Information and Education Technology, Vol. 2, No. 5, pp: 535-537.
40. Mell, P., Scarfone, K., & Romanosky, S. (2007, June). A Complete Guide to the Common Vulnerability Scoring System Version 2.0. In Published by FIRST-forum of Incident Response and Security Teams, pp: 1-24.
41. Available online at: https://www.ncbi.nlm.nih.gov/pmc/articles /PMC6077627/
42. Available online at: https://www.csesoftware.com/data-integrity-for-healthcare/
43. Available online at: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9016269.
44. Available online at: https://www.healthcareit.com.au/opinion/striking-balance-between-data-availability-and-security-healthcare
45. Li, W., & Henry, S. (1993, May). Maintenance Metrics for the Object-oriented Paradigm. In [1993] Proceedings First International Software Metrics Symposium, IEEE, pp. 52 – 60.
46. Flechais, I., Sasse, M. A., & Hailes, S. M. (2003, August). Bringing Security Home: a Process for Developing Secure and Usable Systems. In Proceedings of the 2003 Workshop on New Security Paradigms, pp. 49-57.

47. Available online at: https://blog.thehcigroup.com/continuous_availability_of_clinical_applications.
48. Available online at: https://www.stratus.com/industry/healthcare/
49. Gr, B., Maksimchuk, R., Engel, M., Young, B., Conallen, J., & Houston, K. (2007). Object-Oriented Analysis and Design with Applications. Addision Wesly, 2007.
50. Bishop, Matt. (2002). Computer Security: Art and Science.
51. F. Droma, Project report on "An automated System for Patient Record System", Department of Information Technology Maker Ere University. (2009).
52. Mahmood, A. K. (2010). Information Security Management of Healthcare System.
53. Sohaib, O., Naderpour, M., Hussain, W., & Martinez, L. (2019). Cloud Computing Model Selection for e-commerce Enterprises Using a New 2-tuple Fuzzy Linguistic Decision-making Method. Computers & Industrial Engineering, Vol. 132, 47-58.
54. Chen, J. F., Hsieh, H. N., & Do, Q. H. (2015). Evaluating Teaching Performance based on Fuzzy AHP and Comprehensive Evaluation Approach. Applied Soft Computing, Vol. 28, pp: 100-108.
55. Schmeelk, S. (2020, January). Creating a Standardized Risk Assessment Framework Library for Healthcare Information Technology. In Proceedings of the 53rd Hawaii International Conference on System Sciences.
56. Available Online at: https://books.google. co.in/books?hl=en&lr=&id=ZHez2BXgIeQC&oi=fnd&pg=PP1&dq=software+makes+life+easier&ots=oR9dO204Su&sig=2_8lAIKjou6VwjBns0VY_CTHcOE#v=onepage&q=software%20makes%20life%20easier&f=false.
57. Top 10 Software Development Risks, Available Online at: https://www.itproportal.com/2010/06/14/top-tensoftwaredevelopment-risks/Last.

58. IT Security Vulnerability v/s Threat v/s Risk: What's the Difference?, Available Online at: http://www. bmc.com/ blogs/security-vulnerability-vs-threat-vs-risk-whats-difference/.
59. Mohd. Waris Khan, 2019, A Ph.D., Thesis On Design And Development Of Security Test Case Optimization.
60. Roney, K. (2012). Handle hospital data breaches with care: 5 issues to consider. Becker's Hospital Review, 14.
61. Landolt, S., Hirschel, J., Schlienger, T., Businger, W., & Zbinden, A. M. (2012). Assessing and Comparing Information Security in Swiss Hospitals. Interactive Journal of Medical Research, Vol. 1, No. 2, e2137.
62. Available online at: https://www.healthcarebusinesstech.com/3-healthcare-technology-security-risks-you-should-know-about/.
63. Mehta, D. M. (2007). Effective Software Security Management. OWASP - Open Web Application Security Project.
64. Peterson, G. (2004). Collaboration in a Secure Development Process Part 1. Information Security Bull, Vol. 9, pp: 165-172.
65. Pfleeger, S., & Cunningham, R. (2010). Why Measuring Security is hard. IEEE Security & Privacy, Vol. 8, No. 4, pp: 46-54.
66. McCurley, J., Zubrow, D., & Dekkers, C. (2008). Measures and Measurement for Secure Software Development. Software Engineering Institute.
67. Kaner, C. (2004). Software Engineering Metrics: What do they Measure and how do we know?, IEEE CS.In 10th International Software Metrics Symposium Metrics 2004. IEEE Computer Society Press, 2004.
68. Taylor, D., & McGraw, G. (2005). Adopting a Software Security Improvement Program. IEEE Security & Privacy, Vol. 3, No. 3, pp: 88-91.
69. Chandra, S., & Khan, R. A. (2009, January). Software Security Metric Identification Framework. In Proceedings of the

International Conference on Advances in Computing, Communication and Control, pp: 725-731.

70. Abdulrazeg, A. A., Norwawi, N. M., & Basir, N. (2012, June). Security Metrics to Improve Misuse Case Model. In Proceedings Title: 2012 International Conference on Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), IEEE. pp: 94-99.
71. Jain, S., & Ingle, M. (2014). Security Metrics and Software Development Progression. Journal of Engineering Research and Applications, pp: 2248-9622.
72. Wysopal, C. (2008). Building Security into your Software-Development Lifecycle. SC Mag, 30. Available Online at: http:/www. scmagazineus.com/building-security-into-your-software-develop ment-lifecycle/article/104705. Last visit May 15, 2016.
73. Schultz, Jr, E. E., Brown, D. S., & Longstaff, T. A. (1990). Responding to Computer Security Incidents: Guidelines for Incident Handling (No. UCRL-ID-104689). Lawrence Livermore National Lab., CA (USA). Available Online at: ftp://ftp.cert.dfn.de/pub/docs/csir/ihg.ps.gz. Last visit May 20, 2016.
74. Berander, P., & Jönsson, P. (2006, September). A Goal Question Metric Based Approach for Efficient Measurement Framework Definition. In Proceedings of the 2006 ACM/IEEE International Symposium on Empirical Software Engineering, pp: 316-325.
75. Agarwal, A., & Khan, R. A. (2012). Role of Coupling in Vulnerability Propagation Object Oriented Design Perspective. Software Engineering: An International Journal (SEIJ), Vol. 2, No. 1, pp: 60-68.
76. Abushark, Y. B., Khan, A. I., Alsolami, F. J., Almalawi, A., Alam, M. M., Agrawal, A., & Khan, R. A. (2021). Usability Evaluation Through Fuzzy AHP-TOPSIS Approach: Security Requirement

Perspective. CMC-Computers Materials & Continua, Vol. 68, No. 1, pp: 1203-1218.

77. Kumar, R., Baz, A., Alhakami, H., Alhakami, W., Agrawal, A., & Khan, R. A. (2021). A Hybrid Fuzzy Rule-Based Multi-Criteria Framework for Sustainable-Security Assessment of Web Application. Ain Shams Engineering Journal, Vol. 12, No. 2, pp: 2227-2240.
78. Attaallah, A., Algarni, A., & Khan, R. A. (2021). Managing Security-Risks for Improving Security-Durability of Institutional Web-Applications: Design Perspective. CMC-Computers Materials & Continua, Vol. 66, No. 2, pp: 1849-1865.
79. Al-Zahrani, F. A. (2020). Evaluating the Usable-Security of Healthcare Software through Unified Technique of Fuzzy Logic, ANP and TOPSIS. IEEE Access, Vol. 8, pp: 109905-109916.
80. Lv, Z., & Qiao, L. (2020). Analysis of Healthcare Big Data. Future Generation Computer Systems, Vol. 109, pp: 103-110.
81. Altowaijri, S. M. (2020). An Architecture to Improve the Security of Cloud Computing in the Healthcare Sector. In Smart Infrastructure and Applications, Springer, Cham, pp: 249-266.
82. Kaur, J., Khan, A. I., Abushark, Y. B., Alam, M. M., Khan, S. A., Agrawal, A., & Khan, R. A. (2020). Security Risk Assessment of Healthcare Web Application through Adaptive Neuro-Fuzzy Inference System: A Design Perspective. Risk Management and Healthcare Policy, 13, pp: 355-371.
83. Hathaliya, Jigna., & Tanwar, Sudeep., (2020). An Exhaustive Survey on Security and Privacy Issues in Healthcare 4.0. Computer Communications, Vol. 153. pp: 311-335,
84. Praveena, D., & Rangarajan, P. (2020). A Machine Learning Application for Reducing the Security Risks in Hybrid Cloud Networks. Multimedia Tools and Applications, Vol. 79, No. 7, pp: 5161-5173.

85. Abu-Elezz, I., Hassan, A., Nazeemudeen, A., Househ, M., & Abd-Alrazaq, A. (2020). The Benefits and Threats of Blockchain Technology in Healthcare: A Scoping Review. International Journal of Medical Informatics, Vol. 142, pp: 1-9.
86. Lu, Y., & Sinnott, R. O. (2020). Security and Privacy Solutions for Smart Healthcare Systems. In Innovation in Health Informatics, Academic Press, pp: 189-216.
87. Wang, Z., Gong, L., Yang, J., & Zhang, X. (2020). Cloud Assisted Elliptic Curve Password Authenticated Key Exchange Protocol for Wearable Healthcare Monitoring System. Concurrency and Computation: Practice and Experience, Vol. 107, pp: 1-12.
88. Vijayakumar, K., & Bhuvaneswari, V. (2020, February). A Ubiquitous First Look of IoT Framework for Healthcare Applications. In 2020 International Conference on Emerging Trends in Information Technology and Engineering, IEEE, pp: 1-7.
89. Alhakami, W., Baz, A., Alhakami, H., Pandey, A. K., & Khan, R. A. (2020). Symmetrical Model of Smart Healthcare Data Management: A Cybernetics Perspective. Symmetry, Vol. 12, No. 12, pp: 1-16.
90. Yongjun, T. (2020). Security Design and Application of Internet of Things Based on Asymmetric Encryption Algorithm and Neural Network for COVID-19. Journal of Intelligent & Fuzzy Systems, (Preprint), pp: 1-9.
91. Wortman, P. A., & Chandy, J. A. (2020). SMART: Security Model Adversarial Risk-based Tool for Systems Security Design Evaluation. Journal of Cyber security, Vol. 6, No. 1, pp: 1-8.
92. Agrawal, A., Seh, A. H., Baz, A., Alhakami, H., Alhakami, W., Baz, M., & Khan, R. A. (2020). Software Security Estimation Using The Hybrid Fuzzy ANP-TOPSIS Approach: Design tactics perspective. Symmetry, Vol. 12, No. 4, pp: 1-21.
93. Li, J. (2020). Vulnerabilities Mapping based on OWASP-SANS: a Survey for Static Application Security Testing (SAST). Annals of

Emerging Technologies in Computing (AETiC), Print ISSN, 2516-0281.

94. Zarour, M., Alenezi, M., & Alsarayrah, K. (2020). Software Security Specifications and Design: How Software Engineers and Practitioners Are Mixing Things up. In Proceedings of the Evaluation and Assessment in Software Engineering pp: 451-456.
95. Sha, K., Yang, T. A., Wei, W., & Davari, S. (2020). A survey of Edge Computing-based Designs for IoT Security. Digital Communications and Networks, Vol. 6, No. 2, pp: 195-202.
96. Cilliers, L. (2020). Wearable Devices in Healthcare: Privacy and Information Security Issues. Health information management journal, Vol. 49, No.2, pp: 150-156.
97. Alenezi, M., & Almuairfi, S. (2019). Security risks in the software development lifecycle. International Journal of Recent Technology and Engineering, Vol. 8, No. 3, pp: 7048-7055.
98. Nong, Z., & Gainsbury, S. (2020). Website Design Features: Exploring How Social cues Present in the online environment may Impact Risk Taking. Human Behavior and Emerging technologies, Vol. 2, No. 1, pp: 39-49.
99. Gillespie, A. A., & Magor, S. (2020, February). Tackling online Fraud. In ERA Forum, Vol. 20, No. 3, pp. 439-454. Springer Berlin Heidelberg.
100. Kaur, J., Alka, R., & Khan, A. (2018). Major Software Security Risks at Design Phase. ICIC Express Lett Int J Res Surv. Vol. 12. Pp: 1155-1162.
101. Assal, H., & Chiasson, S. (2018). Security in the software development lifecycle. In Fourteenth Symposium on Usable Privacy and Security ({SOUPS} 2018) pp. 281-296.
102. Shapaval, R., & Matulevičius, R. (2018, July). Towards the Reference Model for Security Risk Management in Internet of

Things. In International Baltic Conference on Databases and Information Systems, Springer, Cham, pp: 58-72.

103. Siddiqui, S. T. (2017). Significance of Security Metrics in Secure Software Development, International Journal of Applied Information Systems (IJAIS) Foundation of Computer Science FCS, New York, USA. Vol. 12, No. 6,
104. Khan, M. W., Pandey, D., & Khan, S. A. (2018). Test Plan Specification using Security Attributes: A Design Perspective. An International Journal of Research and Surveys. ICIC Express Letters, Vol. 12, No. 10, pp: 1061-1069.
105. Khan, M. W., Pandey, D., & Khan, S. A. (2016, August). Critical Review on Software Testing: Security Perspective. In International Conference on Smart Trends for Information Technology and Computer Communications, Springer, Singapore, pp: 714-723.
106. Pandey, S. K., & Batra, M. (2013). Security Testing in Requirements Phase of SDLC. International Journal of Computer Applications, Vol. 68, No. 9, pp: 31-35.
107. Adetoba, B., & Ogundele, I. (2018). Requirements Engineering Techniques in Software Development Life Cycle Methods: A Systematic Literature Review. International Journal of Advanced Research in Computer Engineering & Technology, Vol. 7, No. 10, pp: 733-743.
108. Khan, M. W., Sankhwar, S., & Singh, V. (2016). Security Testing Profile: An Introduction.National conference on Information Security Challenges (NCISC-2016), pp. 47-49.
109. C. Mallow, Authentication Methods and Techniques, web reference:www.giac.org/cissp-paper/2.pdf
110. Web Application Security Fundamentals, Chapter 1, Available online at: https://msdn.microsoft.com/en-us/library/ff648636.aspx, 2017.

111. Available online at https://www.castsoftware.com/research-labs/ software-risk
112. Available online at: https://cwe.mitre.org/data/definitions/767 .html
113. Available online at: https://www.security-database.com/cwe.php? name=CWE-767
114. Available online at: https://cwe.mitre.org/data/definitions/260 .html
115. Available online at: https://www.cybersecurity-help.cz/vdb/ cwe/ 260/.
116. Available online at: https://cwe.mitre.org/data/definitions/311.html.
117. Available online at: https://www.cybersecurity-help.cz/vdb/ SB2020091803.
118. Available online at: https://www.cybersecurity-help.cz/vdb/cwe/ 620/.
119. Available Online at: https://www.cybersecurity-help.cz/vdb/ SB2018092902.
120. Available online at: https://cwe.mitre.org/data/definitions/366.html
121. Available online at: https://cwe.mitre.org/data/definitions/426.html
122. Available online at: https://www.cybersecurity-help.cz/vdb/cwe /426/
123. Available online at: https://www.cvedetails.com/cwe-details/494/ Download-of-Code-Without-Integrity-Check.html
124. Available online at: https://www.cybersecurity-help.cz/vdb/SB 2020061609
125. Available online at: https://cwe.mitre.org/data/definitions/362 .html
126. Available online at: https://www.cvedetails.com/cwe-details /362/Race-Condition.html
127. Available online at: ttps://www.cvedetails.com/cwe-details/454/ External-Initialization-of-Trusted-Variables-or-Data-Stores.html
128. Available online at: Available online at: https://cwe.mitre. org/data/ definitions/454.html
129. Available online at: https://cwe.mitre.org/data/definitions/915 .html

130. Pauli, Joshua & Xu, Dianxiang. (2006 August). Integrating Functional and Security Requirements with Use Case Decomposition.In 11th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS'06), IEEE, pp: 57-66.
131. Walton, G. H., Longstaff, T. A., & Linger, R. C. (2006). Technology Foundations for Computational Evaluation of Software Security Attributes. Carnegie-Mellon University Pittsburgh Pa Software Engineering Institute.
132. Firesmith, D. G. (2003). Security Use Cases. Journal of Object Technology, Vol. 2, No. 3, pp: 53-64.
133. Whitten, A. (2004). Making Security Usable. Unpublished Ph. D. Thesis, CS, CMU.
134. Jain, S., & Ingle, M. (2011). Software Security Requirements Gathering Instrument. International Journal of Advanced Computer Science and Applications, Vol. 2, No. 7. pp: 116-121.
135. Legislative Proposals to Protect Online Privacy and Security, Available online at: https://www.justice.gov/archives/opa/blog/legislative-proposals-protect-online-privacy-and-security
136. Criminalizing the Overseas Sale of Stolen U.S. Financial Information, Available at: https://www.justice.gov/ archives/opa/blog/criminalizing-overseas-sale-stolen-us-financial-information.
137. Prosecuting the Sale of Botnets and Malicious Software, Available Online at: https://www.justice.gov/archives/opa/blog/prosecuting-sale-botnets-and-malicious-software
138. Khan, Suhel & Khan, Prof. Raees. (2013). Software Security Testing Process: Phased Approach. 276, pp: 211-217.
139. Zhang, D., Nie, C., & Xu, B. (2008, November). A Markov Decision Approach to Optimize Testing Profile in Software Testing. In 2008 9th International Conference for Young Computer Scientists, IEEE, pp: 1205-1210.

140. Prasad, R. S., Rao, K. R. H., & Kantha, R. R. L. (2011). Software Reliability Measuring Using Modified Maximum Likelihood Estimation and SPC. International Journal of Computer Applications, Vol. 21, No. 7, pp: 1-5.
141. Sattarova Feruza, Y., & Kim, T. H. (2007). IT Security Review: Privacy, Protection, Access Control, Assurance and System Security. International Journal of Multimedia and Ubiquitous Engineering, Vol. 2, No. 2, pp: 17-32.
142. Kumar, R., Khan, S. A., & Khan, R. A. (2014). Software Security Testing: a Pertinent Framework. Journal of Global Research in Computer Science, Vol. 5, No. 3, pp: 23-27.
143. Brunil, D., Haddad, H. M., & Romero, M. (2009, April). Security Vulnerabilities and Mitigation Strategies for Application Development. In 2009 Sixth International Conference on Information Technology: New Generations, IEEE, pp: 235-240.
144. Available online at: https://www.owasp.org/index.php/Top_10-2017_ Top_10
145. Khan, R. A., & Mustafa, K. (2009). From Threat to Security Indexing: a Causal Chain. Computer Fraud & Security, Vol. 2009, No. 5, pp: 9-12.
146. McGraw, G. (2004). Software security. IEEE Security & Privacy, Vol. 2, No. 2, pp: 80-83.
147. Gu.T.Y. Shi, Y.S., & Fang, Y.U., (2010). Research on Software Security Testing, In: World Academy of Science, Engineering and Technology, pp: 647-651.
148. Anwer, F., Nazir, M., & Mustafa, K. (2017). Security Testing. Trends in Software Testing, Springer Nature. pp: 35-66.
149. Kumar, R., Khan, S. A., & Khan, R. A. (2016). Analytical Network Process for Software Security: A Design Perspective. CSI Transactions on ICT, Vol. 4, No. (2-4), pp: 255-258.

150. Chang, S. H. (2012). Fuzzy multi-criteria evaluation and statistics. Wunan Books, Taipei. Vol. 73, pp: 208-221.
151. Do Chung, B., & Seo, K. K. (2015). A Cloud Service Selection Model based on Analytic Network Process. Indian Journal of Science and Technology, Vol. 8, No. 18, pp: 1-5.
152. Koçak, S. A., Alptekin, G. I., & Bener, A. (2014). Evaluation of Software Product Quality Attributes and Environmental Attributes using ANP Decision Framework. In RE4SuSy@ RE, Conference: 3rd International Workshop on Requirements Engineering for Sustainable Systems, pp: 37-44.
153. Schellnhuber, H. J., & Wenzel, V. (Eds.). (2012). Earth System Analysis: Integrating Science for Sustainability. Springer Science & Business Media. Accessed: Sep. 15, 2019, Available: https://sites.hks.harvard.edu/sed/docs/hjs_esa_environment_0510
154. Ishizaka, A., & Nemery, P. (2013). Multi-criteria Decision Analysis: Methods and Software. John Wiley & Sons.
155. Chang, C. W., Wu, C. R., & Lin, H. L. (2008). Integrating fuzzy Theory and Hierarchy Concepts to Evaluate Software Quality. Software Quality Journal, Vol. 16, No. 2, pp: 263-276.
156. Paradis, R., & Tran, B. (2010). Balancing Security/Safety and Sustainability Objectives. National Institute of Building Sciences. Available online at: https://www.wbdg.org/resources/ balancing-security-safety-and-sustainability-objectives.
157. Saaty, T. L. (1990). How to Make a Decision: The Analytic Hierarchy Process. European Journal of Operational Research, Vol. 48, No. 1, pp: 9-26.
158. Dawood, K. A., Sharif, K. Y., Zaidan, A. A., Abd Ghani, A. A., Zulzalil, H. B., andZaidan, B. B. (2019). Mapping and Analysis of Open-Source Software (OSS) Usability for Sustainable OSS Product. IEEE Access, Vol. 7, pp: 65913-65933.

159. Zadeh, L. A. (1996). Fuzzy sets. In Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A Zadeh: pp: 394-432.
160. Chen, J. F., Hsieh, H. N., & Do, Q. H. (2015). Evaluating Teaching Performance based on Fuzzy AHP and Comprehensive Evaluation Approach. Applied Soft Computing, Vol. 28, pp: 100-108.
161. Chou, Y. C., Yen, H. Y., Dang, V. T., & Sun, C. C. (2019). Assessing the Human Resource in Science and Technology for Asian Countries: Application of fuzzy AHP and fuzzy TOPSIS. Symmetry, Vol. 11, No. 2, pp: 251-262.
162. Carter, J. (2012). Coupling and Cohesion: A View of Software Design from the Inside Out. EHR Science, Nov, 12. Online Available online at: https://www.ehrscience.com/2012/11/12/coupling-and-cohesion-a-view-of-software-design-from-the-inside-out-2/.

Annexure - A

Questionnaire Form for Estimating the Impact of Security Risk

Details and Description: Information security is critical consideration for the integration and communication with healthcare systems when sharing private medical information. To evaluate the impact of security risk for secure and trustworthy healthcare web application, a questionnaire/feedback form has been developed. Feedback is required from domain experts in the healthcare management sectors. The methodology is based upon Multi Criteria Decision Analysis Methods.

Your suggestions will surely help to improve the methodology. So you are requested to kindly give your opinion for the given set of questionnaire. The scale for answers has been given in the table 1. The responses are to be given in numeric form. The reciprocal numeric values represent the opposite of the importance level.

Table 1: Scale of Linguistic Values with Numerical Values

S. No.	Linguistic Values	Numeric Values	Reciprocal Values
1	Equal Important (Eq)	1	1
2	Intermediate Value between Equal and Weekly (E & W)	2	2^{-1}
3	Weekly Important (WI)	3	3^{-1}
4	Intermediate Value between Weekly and Essential (W & E)	4	4^{-1}
5	Essential Important (EI)	5	5^{-1}
6	Intermediate Value between Essential and Very Strongly (E & VS)	6	6^{-1}
7	Very Strongly Important (VS)	7	7^{-1}
8	Intermediate Value between Very Strongly and Extremely (VS & ES)	8	8^{-1}
9	Extremely Important (ES)	9	9^{-1}

Please read the following questions and put check marks on the pair wise comparison matrices. If a criteria on the left is more important than the matching one on the right, put your check mark to the left of the importance ''Equal (1)'' under the importance level you prefer. If a criteria on the left is less important than the matching one on the right, put your check mark to the right of the importance 'Equal (1)' under the importance level you. Reciprocal value means the opposite effect of the factor of assigned value. Here, total five groups are available. Please put your mark for e ach group.

Importance of One Criteria Over Another																	
Q. N.		9	8	7	6	5	4	3	2	1	2^{-1}	3^{-1}	4^{-1}	5^{-1}	6^{-1}	7^{-1}	8^{-1}	9^{-1}	
1	T1							✓											T2
2	T1													✓					T3
3	T1											✓		✓					T4
4	T1								✓										T5
5	T2					✓													T3
6	T2			✓															T4
7	T2															✓			T5
8	T3													✓					T4
9	T3							✓											T5
10	T4							✓											T5

Importance of One Criteria Over Another																			
Q.N.		9	8	7	6	5	4	3	2	1	2^{-1}	3^{-1}	4^{-1}	5^{-1}	6^{-1}	7^{-1}	8^{-1}	9^{-1}	
1	T11											✓							T12
2	T11					✓													T13
3	T12							✓											T13

Importance of One Criteria Over Another																		
Q.N.		9	8	7	6	5	4	3	2	1	2^{-1}	3^{-1}	4^{-1}	5^{-1}	6^{-1}	7^{-1}	8^{-1}	9^{-1}	
1	T21							✓											T22
2	T21					✓													T23
3	T21											✓							T24
4	T21											✓							T25
5	T21															✓			T26
6	T21													✓					T27
7	T21												✓						T28
8	T22							✓											T23
9	T22					✓													T24
10	T22							✓											T25
11	T22							✓											T26
12	T22										✓								T27
13	T22													✓					T28
14	T23												✓						T24
15	T23								✓										T25
16	T23							✓											T26
17	T23					✓													T27
18	T23			✓															T28
19	T24												✓						T25
20	T24															✓			T26
21	T24												✓						T27
22	T24														✓				T28
23	T25																		T26
24	T25											✓							T27
25	T25								✓										T28
26	T26						✓												T27
27	T26							✓											T28
28	T27												✓						T28

Importance of One Criteria Over Another																		
Q.N.		9	8	7	6	5	4	3	2	1	2^{-1}	3^{-1}	4^{-1}	5^{-1}	6^{-1}	7^{-1}	8^{-1}	9^{-1}	
1	T41						✓												T42
2	T41											✓							T43
3	T41			✓															T44
4	T42													✓					T43
5	T42										✓								T44
6	T43							✓											T44

Importance of One Criteria Over Another																		
Q.N.		9	8	7	6	5	4	3	2	1	2^{-1}	3^{-1}	4^{-1}	5^{-1}	6^{-1}	7^{-1}	8^{-1}	9^{-1}	
1	T51						✓												T52

Your Comments (Please mark corrections as and where required): Please find details in E-Mail:

Expert's Name and Signature: Dr. Waris Khan

Please return this to: Syed Anas Ansar (syed000anas@gmail.com), Department of Information Technology, SIST, Babasaheb Bhimrao Ambedkar University, Lucknow, Uttar Pradesh, India. 226025

Annexure – B

Form for Rating the Security Risk Framework

Details and *Description*: Information security is critical consideration for the integration and communication with healthcare systems when sharing private medical information. To evaluate the impact of security risk for secure and trustworthy healthcare web application, a questionnaire/feedback form has been developed. Feedback is required from domain experts in the healthcare management sectors. The methodology is based upon Multi Criteria Decision Analysis Methods. A computational methodology has been proposed to evaluate the impact of security risk on healthcare web application. To demonstrate the same, the researcher has taken ratings of various attributes of security. Further, ratings of the attributes may be helpful for researcher to evaluate the crucial impact.

Your ratings for attributes will surely help to improve the methodology. So you are requested to kindly give your opinion for the given attributes. The scale for answers has been given in the Table 1. The responses are to be given in numeric form.

Table 1: Rating Scale

S. No.	Linguistic Value	Numeric Value of Ratings
1	Very Low (VL)	0.1
2	Low (L)	0.3
3	Medium (M)	0.7
4	High (H)	0.9
5	Very High (VH)	1.0

You need to rate the different attributes with the given scale. The ratings have been divided in five parts including 0.1 (lowest rating) while 1.0 (highest rating) of the attributes.

S. No.	Name of the Security Risk Attributes	Ratings
1	Access to critical Private Variable via Public Method	M
2	Password in Configuration	H
3	Missing Encryption of Sensitive Data	H
4	Unverified Password Change	H
5	Race Condition within a Thread	L
6	Untrusted Search Path	VL
7	Download of Code Without Integrity Check	L
8	Concurrent execution using shared resource with improper synchronization ('Race Condition')	L
9	External Initialization of Trusted Variables or data stores	M
10	Improperly Controlled Modification of Dynamically-Determined Object Attributes	L

Your Comments (Please mark corrections as and where required):
Please find details in E-Mail:

—

Expert's Name and Signature: Dr. Waris Khan Khan

Please return this to: Syed Anas Ansar (syed000anas@gmail.com), Department of Information Technology, SIST, Babasaheb Bhimrao Ambedkar University, Lucknow, Uttar Pradesh, India. 226025

Annexure – C

SAQ Infosys GSTN .09AADFQ2053Q1Z8

Dated: 10th February 2021

TO WHOMSOEVER IT MAY CONCERN

This is to certify that **Mr. Syed Anas Ansar**, a Ph.D Scholar from Department of Information Technology, Babasaheb Bhimrao Ambedkar University (A Central University), Lucknow, India, has conducted some security risk assessment work with our company. He has used our project details along with the other details for research purposes.

The identification of these projects has been concealed, as per our desire and company policy. The source data that is going into thesis is correct to the best of my knowledge and belief.

Authorized Signatory

SAQ Infosys Cell- 09208685090 Sector 4/250, Vikas Nagar Lucknow 226022

Sector- 4/250 Vikasnagar Lucknow - 226022 Cell– +919918200140,+919335611550 E-mail: saqinfosys@gmail.com