

IoT security protocol to ensure authentication, authorization and scalability

**P . Maurakirinathan
2019**



IoT security protocol to ensure authentication, authorization and scalability

**A dissertation submitted for the Degree of Master of
Computer Science**

**P . Maurakirinathan
University of Colombo School of Computing
2019**



Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: Paramalingam Maurakirinathan

Registration Number: 2016/MCS/070

Index Number:16440701

P. Maurakirinathan

Signature:

Date: 29/04/2019

This is to certify that this thesis is based on the work of Mr. P . Maurakirinathan under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by:

Supervisor Name: Dr. Kasun de zoysa

Signature:

Date:

Abstract

IoT technology is vastly being used by the users these days because of the technologies which have emerged with the improvement of technology network and distributed computing. Currently people using IoT devices for day today activities to make life easier. IoT services providers and IoT servers help peoples to pass messages and actions to devices.

When user need to pass action or message to device, the processes goes through the cyber supply chain whereas the cyber supply chain security is still in a preliminary stage, where there's a possible chance of hackers breaking in and stealing the messages or action and other data such as device information. Due to this reason, the user's data confidentiality would be lost since the third-party server will know more than our action or messages and the content of data. To solve this problem an DNS based protocol is proposed in order to provide security for IoT devices messages and actions.

Acknowledgement

The hard work which was put in by me, in finishing this thesis has been a very tough path throughout but at the same time it was a pleasant experience. It would have never been possible without the immense courage, assistance and support from the people who were with me throughout the project. I would like to expand my wholehearted thanks to each and every one of them.

I am highly grateful to my supervisor Dr. T.N. K. De Zoysa, for his persistent guidance and supervision. Should be also thankful for encouraging with ideas in order to endeavour to greater heights and to achieve the fullest prospective.

I would like to convey my heartiest gratitude to everyone who have helped me by taking in part in the surveys, interviews and providing their views for the requirements gathering and the critical evaluation chapters.

Eventually I would like to indicate my sincere appreciation to my friends for their essence co-operation and motivation which guided me to finish the thesis successfully.

Table of Contents

List of Figures.....	8
List of Tables.....	9
List of Abbreviation.....	10
Chapter 1: Introduction.....	11
1.1 Chapter Overview.....	12
1.2 Problem Domain.....	12
1.3 Previous Work.....	12
1.4 Project Aim.....	13
1.5 Project Objectives.....	14
1.5 Feature of the Prototype.....	14
Chapter 2: Literature Review.....	15
2.1 Chapter Overview.....	16
2.2 Problem Domain.....	16
2.3 IoT and IoT Security.....	16
2.3.1 IoT Technology.....	16
2.3.2 How IoT works.....	17
2.3.3 Benefits of IoT.....	18
2.3.4 Consumer and enterprise IoT applications.....	19
2.3.5 IoT Security Risks and Threats.....	19
2.3.6 Types of IoT Security Attacks.....	20
2.4 Existing Solution For IoT security.....	21
2.4.1 DNS and the Internet of Things Outlining the challenges faced by DNS in the Internet of Things.....	21
2.4.2 Internet of things(IoT) Security Best Practices.....	22
2.5 Overall summary of Existing Solutions.....	23
2.6 Chapter Summary.....	23
Chapter 3: Project Management.....	24
3.1 Chapter Overview.....	25
3.2 Project Management Methodology.....	25
3.3 Development Methodology.....	25
3.4 Research Methodology.....	26
3.5 Chapter Summary.....	27
Chapter 4: System Architecture & Design.....	28
4.1 Chapter Overview.....	29
4.2 High Level Design.....	29
4.2.1 Rich Picture of the IoT Security System.....	29
4.2.1.2Server side.....	31
4.2.2 High Level Architecture.....	32
4.2.3 Three-tier architecture.....	32
4.2.4 High Level Architecture Diagram.....	32
4.3 System Design.....	33
4.3.1 Selection of Design Methodology.....	33
4.6 Design Goals for Overall Solution.....	34
4.7 Chapter Summary.....	34
Chapter 5: Implementation.....	35

5.1 Chapter Overview.....	36
5.2 Technology Selections.....	36
5.2.1 Selection of programming languages.....	36
5.2.2 Web service exposure approach.....	36
5.2.4 Selection of an IDE and a deployment environment.....	37
5.2.5 Selection of Third-party server.....	37
5.3 Implementation of REST service.....	38
5.3.1 Implementation CURD API of user, device, location services.....	38
5.4 Chapter Summary.....	39
Chapter 6: Testing.....	41
6.1 Chapter Overview.....	42
6.2 Objectives and Goals of Testing.....	42
6.3 Testing Criteria.....	42
6.4 White Box And Black Box Testing.....	43
6.4.1 White Box Testing.....	43
6.4.2 Black Box Testing.....	43
6.5 Functional Requirements Testing.....	43
6.6 Non Functional Requirements Testing.....	44
6.6.1 Accuracy Testing.....	44
6.6.2 Performance Testing.....	46
6.6.3. Load and Scalability Testing.....	48
6.7 Limitations of the Testing Process.....	49
6.8 Chapter Summary.....	49
Chapter 7: Evaluation.....	50
7.1 Chapter Overview.....	51
7.2 Evaluation Plan.....	51
7.3 Evaluation Criteria.....	51
Table 7.1 – Evaluation Criteria.....	52
7.4 Evaluation Methodology and Approach.....	52
7.5 Chapter Summary.....	53
Chapter 8: Conclusion.....	54
8.1 Chapter Overview.....	55
8.2 Achievement of Aim and Objectives.....	55
8.2.1 Aim.....	55
8.2.2 Objectives.....	55
8.3 Problems and Challenges Faced.....	58
8.4 Limitations of the Research.....	58
8.6 Future Enhancements.....	59
8.7 Contribution.....	59
8.8 Concluding Remarks.....	60
References.....	61

List of Figures

Figure 2.1- IoT Technology.....	17
Figure 2.2 - How IoT works.....	17
Figure 2.3 - Benefits of IoT.....	18
Figure 2.4- IoT Applications.....	19
Figure 3.1 - comparison of development mythologies.....	26
Figure 4.1 -Client side Rich Picture.....	29
Figure 4.2 -JWT Token.....	30
Figure 4.3- Server side Rich Picture.....	31
Figure 4.4 - High Level Architecture Diagram.....	32
Figure 6.1- Accuracy Testing Chart.....	46
Figure 6.2 – Performance Testing Chart.....	47
Figure 7.1 – Evaluation Plan	51

List of Tables

Table 2.1 -Access control policies via DNS resolving mechanism advantages and disadvantages...	21
Table 2.2 -Proposed (IoT) security best practice advantages and disadvantages.....	22
Table 6.1- Testing method for functional requirement.....	44
Table 6.2- Accuracy Testing.....	45
Table 6.3- Performance Testing Results.....	47
Table 7.1 – Evaluation Criteria.....	52

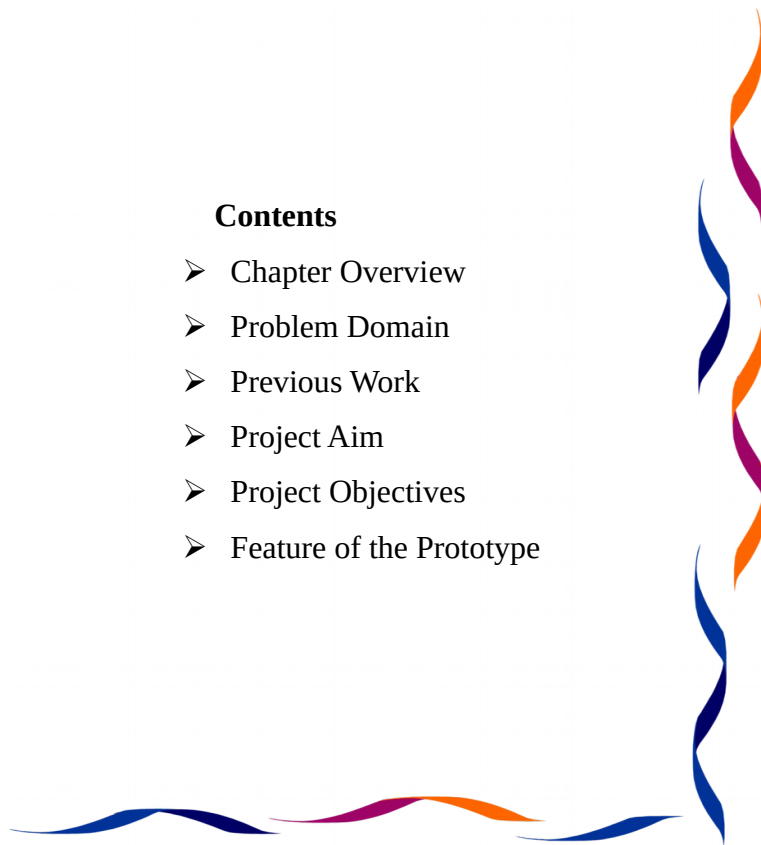
List of Abbreviation

Abbreviation	Definition
IoT	Internet of Things
UDP	User Datagram Protocol
TCP	Transmission Control Protocol
JSON	JavaScript Object Notation
JWT	JSON Web Token
OS	Operating System
DNS	Domain Name System
DB	Database
HMAC	Hash-based message authentication code
SSAD	Structured Systems Analysis and Design
HTTP	Hypertext Transfer Protocol
OOP	Object Oriented Programming
SOAP	Simple Object Access Protocol
REST	Representational State Transfer
OOAD	Object Oriented Analysis and Design

Chapter 1: Introduction

Contents

- Chapter Overview
- Problem Domain
- Previous Work
- Project Aim
- Project Objectives
- Feature of the Prototype



1.1 Chapter Overview

The purpose of this chapter is to give a brief idea about this research project. This report starts with the problem domain which is about in vastly growing IoT technology, securing the peoples' messages and actions through IoT devices using authentication, authorization and scalability. This chapter also includes the previously work done on encrypted data and the aim of this project with objectives as well. Furthermore, features of the prototype, resource requirement and brief chapter summery are being included in this report.

1.2 Problem Domain

In this modern world, several new technologies have emerged with development of network technology and distributed computing. Internet of Things is taking a big part of it and huge amount of data passing and processing through the Internet for IoT devices.”In September 2016, the world witnessed its largest ever IoT botnet attack through Mirai; a string of malicious code which, through the co-opting of vulnerable devices, brought down a swathe of ISPs and online services affecting businesses and consumers alike. The root cause was traced to devices using factory set default usernames and passwords.” (“Canonical”, 2018). Most of the IoT projects are unsecure, due to most of the device requests and responses are sent without security (Hamzic et al., 2016). Therefore in this case secure connection between devices and servers which are connected through internet must be provided. But each and every time, checking the security makes the server busy. So need to find a solution for this to provide authentication, authorization and scalability about requests and responses for devices which are connect through the internet.

1.3 Previous Work

Internet of Things(IoT) is popular throughout the last decade because it has emerged with the development of network technology and distributed computing(“Baseline Security Recommendations for IoT” ,2018).“Internet of Things (IoT): a wired or wireless network of uniquely identifiable connected devices which are able to process data and communicate with each other with or without human involvement.”(Corser et al., 2017). But still the IoT security is in the researching stage and also each messages and actions go through the cyberspace. But cyber supply chain security is still in preliminary stage (“Baseline Security Recommendations for IoT” ,2018).

“A functional naming and service identification method is an essential part in making the IoT global, and DNS is the current method of naming devices on the Internet” which was proposed by Hamzic et al., 2016. It also looks into some challenges DNS will encounter, namely functionality, security and availability. This is a popular method for enforcing access control policies via DNS resolving mechanism (Hamzic et al., 2016). “Internet of things (IoT) Security Best Practices” It’s a research to identify Internet of things security best practices which was proposed by Corser et al. (2017). This research study discuss about “To protect consumers and the public, technical professionals and policy-makers must define and encourage proper security practices. At present, because different precautions are appropriate under different conditions, it is not possible to specify a set of universal rules for IoT security. However, it is possible to describe a set of general principles, or best practices” (Corser et al., 2017). Followings are some of the drawbacks of the above solutions:

- Wasting processing power, memory and power resources.
- DNS DNSSEC does not protect against DDoS attacks
- The servers can only respond for DNS query if the devices are online
- In devices RFID and MAC address can be attacked by Tag cloning, Tag tracking and etc.
- DDoS attacks in network layers can’t be protected by authentication and encryption

Due to the above drawbacks planned to propose a proper solution with the following features

- Authentication
- Authorization
- Scalability

1.4 Project Aim

To design, develop and evaluate a protocol for Internet of Things; To pass messages and actions through IoT devices, with authentication, authorization and scalability.

Further elaborating the project’s aim, there are some existing solutions for this problem. But most of them are not accurate with current internet technology (Corser et al., 2017). Therefore through this protocol the user will be able to pass messages to IoT devices with authentication, authorization and scalability.

The proposed system would support devices which are connected between them or connected to internet with a minor configuration. The proposed system will not cover security layers of third party servers, third party servers encryption & decryption algorithms

1.5 Project Objectives

- IoT and IoT security - to understand how IoT and IoT security works.
- Existing IoT security methodologies - to understand how existing methodologies works and analyze their techniques along with their strengths and weaknesses
- Compare with the existing IoT security methodologies - which were used by other technical experts to provide security for IoT.
- Through this proposed protocol the user will be able to pass messages to IoT devices without losing any Secrecy and confidentiality.

1.5 Feature of the Prototype

The main function of this protocol enable authentication and authorization without leaking any kind of information such as messages and actions

As an example if a user pass messages like switch on bulb it will travel through this protocol without leaking any information

But in this solution

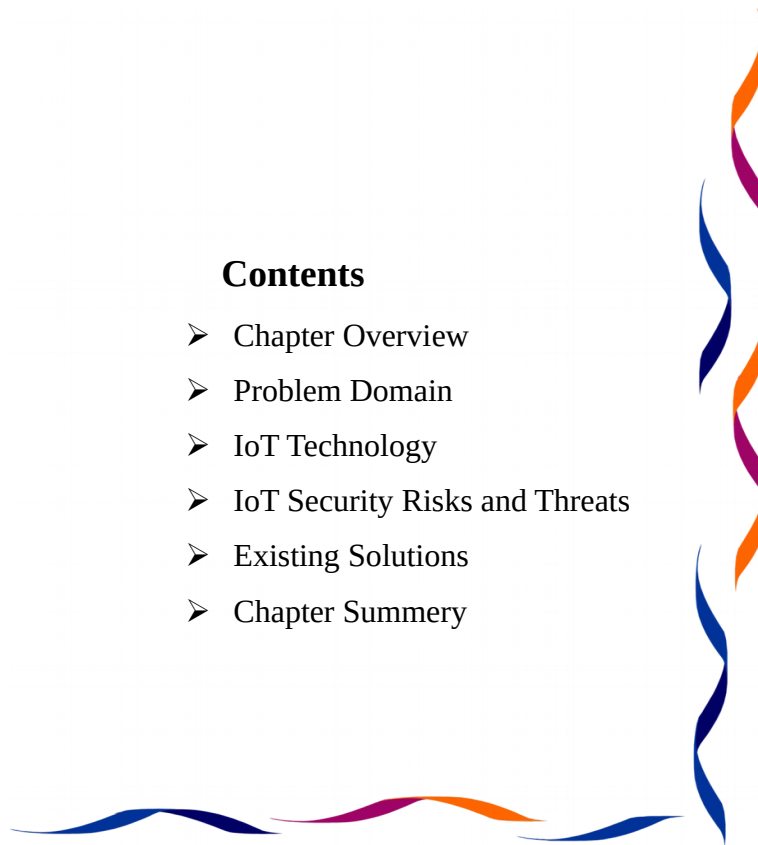
- 1) Checks the user authentication
- 2) Checks the user authorization
- 3) Converts the user's action and messages to cipher text
- 4) Pass cipher messages through protocol

The proposed system would support devices which are connected between them or connected to internet with a minor configuration. The proposed system will not cover security layers of third party servers, third party servers encryption & decryption algorithms.

Chapter 2: Literature Review

Contents

- Chapter Overview
- Problem Domain
- IoT Technology
- IoT Security Risks and Threats
- Existing Solutions
- Chapter Summery



2.1 Chapter Overview

Previous chapter, an introduction in to the project. It talks about the project background with previous work done regarding IoT security with authentication, authorization and scalability. It also provides the project aim and objective, elaborating what is this project and actually how it's going to be done. Here, the thesis discusses about, why it needs securing the peoples' messages and actions through IoT devices using authentication, authorization and scalability along with a critical evaluation. Finally it will be analyzing what have been already done along with the related work on the problem domain.

2.2 Problem Domain

As mentioned in the previous chapter, Most of the IoT projects are unsecured, due to most of the device requests and responses are sent without security (Hamzic et al., 2016). Therefore in this case secure connection between devices and servers which are connected through internet must be provided. But each and every time, checking the security makes the server busy. So need to find a solution for this to provide authentication, authorization and scalability about requests and responses for devices which are connect through the internet.

2.3 IoT and IoT Security

2.3.1 IoT Technology

According to Techtargget article “The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.”(IoT Agenda. 2018. What is internet of things)

Throughout the past decade IoT has become the most popular technology since most of the people from the twenty first century do activities related to devices which are connected to Internet such as Ip cameras , home alarm systems , fire alert system and etc.

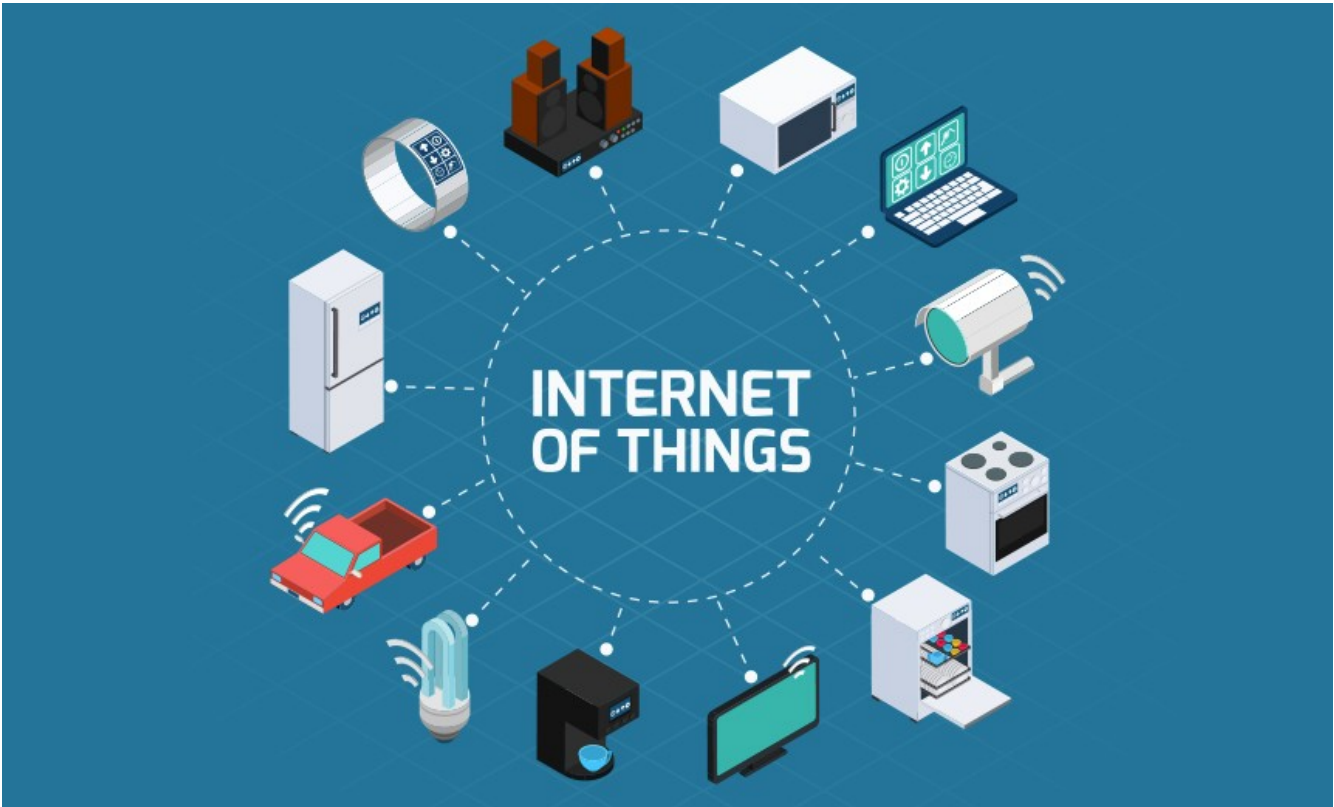


Figure 2.1-IoT Technology (GAO Assesses IoT Vulnerabilities - BankInfoSecurity. 2019)

2.3.2 How IoT works

The smart devices which contains embedded processor, sensors and communication hardware (WiFi, Bluetooth etc) to receive ,send and make actions on devices which connect between them or hub

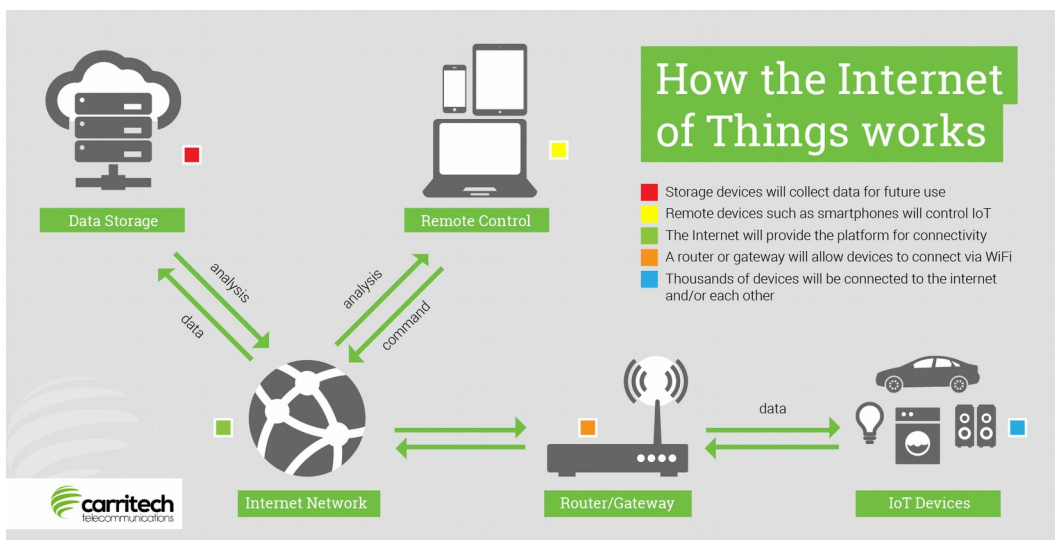


Figure 2.2 - How IoT works (Carritech Telecommunications. 2018)

2.3.3 Benefits of IoT

The IoT offers a number of benefits to organizations and humans, enabling them to:

- monitor their overall business processes.
- improve the customer experience.
- save time and money.
- enhance employee productivity.
- integrate and adapt business models.
- make better business decisions.
- generate more revenue.

(IoT Agenda. 2018. What is internet of things)

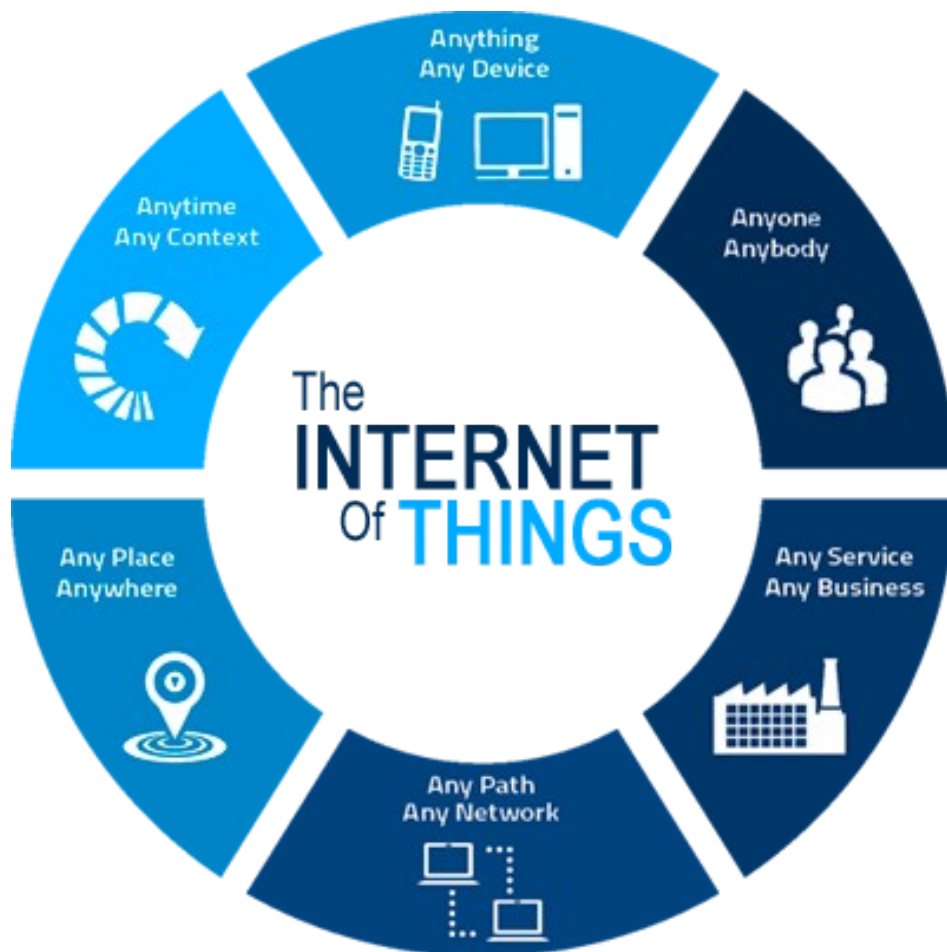


Figure 2.3 - Benefits of IoT (SharpNode Blog | Latest in IoT. 2018.)

2.3.4 Consumer and enterprise IoT applications

In current internet world most of the field using Internet of Things for day today activities. Smart home and industrial IoT (IioT) taking huge part of IoT world (IoT Agenda. 2018. What is internet of things).

following image displays more IoT appliances in current world

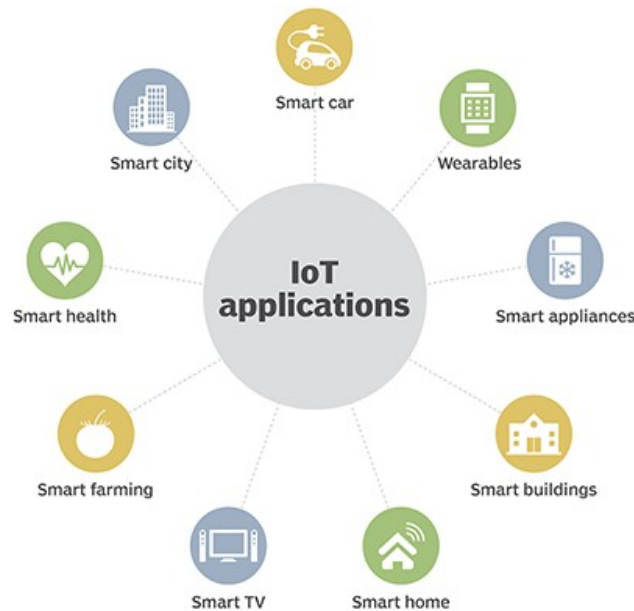


Figure 2.4- IoT Applications (IoT Agenda. 2018. What is internet of things)

2.3.5 IoT Security Risks and Threats

There are so many IoT devices using day today life but we cant make sure it is fully safe because according to Canonical Defining IoT Business Model “Over 50% of IoT devices are unsecure “ (“Canonical”, 2017). and also following examples shows security risks on current IoT

- Stuxnet(2010-2014)- Many experts believe that Stuxnet destroyed up to 1,000 devices
- Mirai botnet(2016)- DDoS attack.
- Cold in Finland(November 2016)- DDoS attack;
- Brickerbot(2016) – DDoS attack
- The botnet barrage(2018)- more than 5,000 IoT devices attacked.
(“Baseline Security Recommendations for IoT” ,2018)

According to Security Intelligence(2019) The security requirement of an internet of things system are complex because still IoT architecture not standardized. Most of the companies and services providers using them own architecture.

Most of the IoT architecture has three layer architecture those are

- Sensor layer
- Network layer
- Application layer

according to above layer IoT attacks also categorized following way

- Sensor layer :- physical attacks
- Network layer:- network attacks
- Application layer :-software attacks(Security Intelligence,2019)

2.3.6 Types of IoT Security Attacks

A. Physical Attacks

if system get attacked by physically with hardware devices in the system it call physical attacks.

- Node Tampering
- RF Interference on RFIDs
- Node Jamming in WSNs
- Malicious Node Injection
- Physical Damage
- Social Engineering
- Sleep Deprivation Attack
- Malicious Code Injection (J Deogirikar *et al* ,2017)

B. Network Attacks

The attacks which are focus on the network of IoT system.

- Traffic Analysis Attacks
- RFID Spoofing
- RFID Cloning
- RFID Unauthorized Access

- Sinkhole Attack
- Man in the Middle Attacks
- Denial of Service
- Routing Information Attacks
- Sybil Attack(J Deogirikar *et al* ,2017)

C. Software Attacks

attacks which are focus on application layer and also attacker attacks system by using virus,worms, spyware,adware etc.

- Phishing Attacks
- Virus, Worms, Trojan horse, Spyware and Aware
- Malicious Scripts
- Denial of Service(J Deogirikar *et al* ,2017)

2.4 Existing Solution For IoT security

2.4.1 DNS and the Internet of Things Outlining the challenges faced by DNS in the Internet of Things

“A functional naming and service identification method is an essential part in making the IoT global, and DNS is the current method of naming devices on the Internet” which was proposed by Hamzic *et al.*, 2016. It also looks into some challenges DNS will encounter, namely functionality, security and availability. This is a popular method for enforcing access control policies via DNS resolving mechanism (Hamzic *et al.*, 2016).

Access control policies via DNS resolving mechanism advantages and disadvantages

Advantages	Disadvantages
access control policies via DNS	Wasting processing power, memory and power resources.
Easy to find device location resolving DNS	DNSSEC does not protect against DDoS attacks
Support with mDNS and DNS-SD	The servers can only respond for DNS query if

	the devices are online
--	------------------------

Table 2.1– Access control policies via DNS resolving mechanism advantages and disadvantages (Hamzic *et al.*, 2016)

2.4.2 Internet of things(IoT) Security Best Practices

It's a research to identify Internet of things security best practices which was proposed by Corser *et al.* (2017). This research study discuss about “To protect consumers and the public, technical professionals and policy-makers must define and encourage proper security practices. At present,because different precautions are appropriate under different conditions, it is not possible to specify a set of universal rules for IoT security. However, it is possible to describe a set of general principles, or best practices” (Corser *et al.*, 2017).

Proposed (IoT) Securitiy Best Prectice advantages and disadvantages

Advantages	Disadvantages
Securing devices with,make hardware tamper resistant,provide for firmware updates/patches ,perform dynamic testing,specify procedures to protect data on device disposal	In devices RFID and MAC address can be attacked by Tag cloning, Tag tracking and etc.
Securing network with strong authentication, use encryption and secure protocol	DDos attacks in network layers can't be protected by authentication and encryption

Table 2.2 - Proposed (IoT) security best practice advantages and disadvantages (Corser *et al.*, 2017)

Apart from the discussed above studies, several other suggestions which are listed below have been given by researchers. “Over 50% of IoT devices are insecure, believe the majority of IoT professionals asked” (Canonical defining IoT, 2017). Canonical defining IoT (2017) is also researching to modify and add some methodology to Ubuntu core OS for IoT security. Research study by Zhou *et al.*, (2018) discussed “The Effect of IoT New Features on Security and Privacy: New Threats, Existing Solutions, and Challenges Yet to Be Solved” (Zhou *et al.*, 2018). Although there are so many mechanisms to provide security to IoT in the current internet world, the above mentioned drawbacks must be solved through building a proper security system for end to end message transfer between devices which are connected between them and Internet.

2.5 Overall summary of Existing Solutions

The existing solutions which were mentioned above solve some part of the identified problem, but none of them assure the security. All of the above solutions have almost the same drawbacks. Some solutions need to add an extra file to the device, which means that it will take some extra space from the device, where the user might not like it. Therefore in this case secure connection between devices and servers which are connected through internet must be provided. But each and every time, checking the security makes the server busy. Then the above solutions take more time and would not work with low bandwidth internet connection .Due to the above reasons, this proposed solution enables the user to provide authentication, authorization and scalability about requests and responses for devices which are connect through the internet

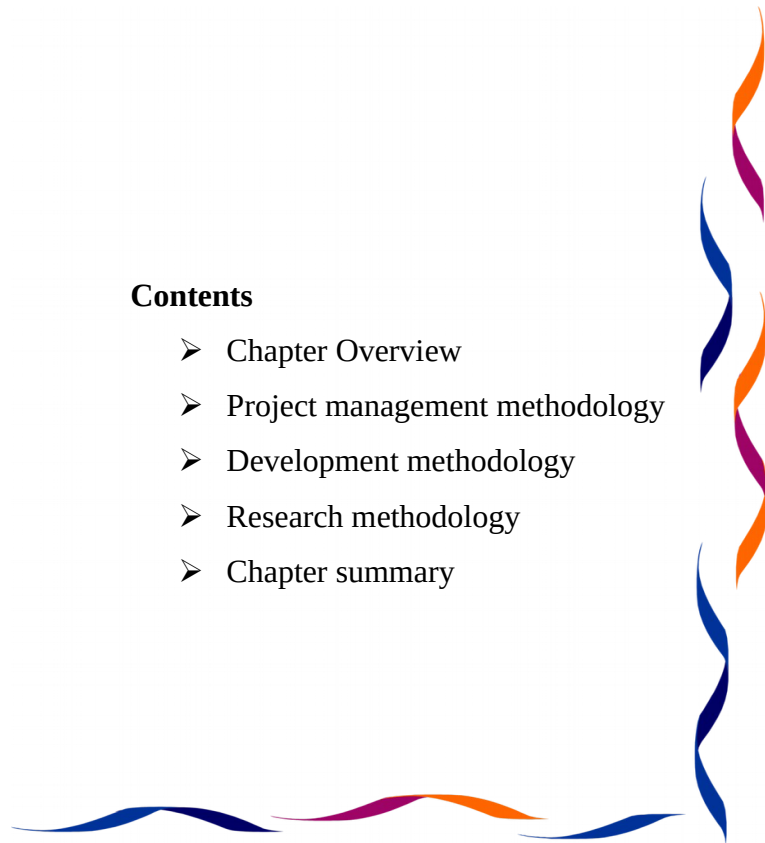
2.6 Chapter Summary

This chapter discussed about, why IoT needs this proper security solution. This chapter starts with explaining about IoT, such as the IoT technologies and the benefits. After that it discusses about the IoT architecture. Next, the chapter focused on what the IoT security is and why the IoT security is important. After that three existing proposed solutions were compared and discussed with advantages and disadvantages. Next chapter will be discussing the project management.

Chapter 3: Project Management

Contents

- Chapter Overview
- Project management methodology
- Development methodology
- Research methodology
- Chapter summary



3.1 Chapter Overview

Previous chapter discussed about the literature review of the project. This chapter will be focusing on the project management process of the proposed system. This starts with finding out a proper project management methodology. Then the chapter moves on to talk about the time and resource allocation. The next chapter will be discussing about the project risks and focuses on the selection of development methodology for this system.

3.2 Project Management Methodology

The PRINCE2 management technology is being used for the proposed system due to the following reasons. This project's scope is big enough so that it takes immense time to research. Important objective of the project is to give a quality output. For this proposed system, time management and quality management are hard to achieve due to the depth of scope. After comparing with the other project management methodologies, PRINCE2 was selected because it's used widely and it's easy to handle time management.

3.3 Development Methodology

To develop and deliver a quality project with low cost and within an actual time period, a proper development methodology is crucial. Due to the above reason, the author has analyzed various kinds of development methodologies such as code and fix, waterfall, Rup, Agile and spiral with mentioning all their characteristics, advantages and disadvantages in order to choose a proper methodology for this project. The following table (Figure 3.1) shows the comparison of development methodologies.

Software development methodologies



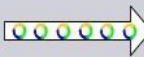

	Code and Fix	Waterfall	RUP	Agile	Spiral
Description	<ul style="list-style-type: none"> Least formal model The coder uses whatever informal design, code and test tools he/she chooses The requirements may or may not be formally written 	<ul style="list-style-type: none"> Follows a strict structure of phases <ul style="list-style-type: none"> Requirements Gathering Design Development Coding Testing Implementation Each phase requires the previous phase to be complete 	<ul style="list-style-type: none"> Rational Unified Model or RUP is structured but iterative Each iterative step involves four phases <ul style="list-style-type: none"> Inception Elaboration Construction Transition Iterations overlap Each iteration outputs a working solution that is incremental to the previous 	<ul style="list-style-type: none"> An expert development team produced a minimal application with limited feature sets are developed delivered in sprints, short working spans of 2-4 weeks Releases are iterative and necessarily incremental 	<ul style="list-style-type: none"> Iterative and focused on risk mitigation Each iterative step involves four phases <ul style="list-style-type: none"> Planning Evaluation Risk analysis Engineering The 
Key Characteristics	<ul style="list-style-type: none"> Unstructured Informal 	<ul style="list-style-type: none"> Structured Linear and simple 	<ul style="list-style-type: none"> Structured Evolutionary 	<ul style="list-style-type: none"> Iterative Incremental Lean 	<ul style="list-style-type: none"> Iterative Low risk
Advantages	<ul style="list-style-type: none"> There is no process overhead There is no training or expertise necessary 	<ul style="list-style-type: none"> Easy to manage large and complex projects when the requirements and scope can be clearly defined and don't keep changing Can be outsourced Produces robust documentation 	<ul style="list-style-type: none"> As changes can be pushed into future releases there is less time pressure on development Risk is managed and exposed early Software evolves and is not produced in one huge effort 	<ul style="list-style-type: none"> Open to change in requirement and scope Flexible Works when requirements are unclear at the beginning of the project 	<ul style="list-style-type: none"> Risk mitigation leads to increased success rate Strong approval documentation and control Great for mission critical projects
Disadvantages	<ul style="list-style-type: none"> There is no planning and control over the development process Leads to spaghetti code 	<ul style="list-style-type: none"> Requirements and design changes cause complexity lead to a lot of rework Success of the project depends on the correctness and completeness of each phase 	<ul style="list-style-type: none"> The approach is complex and can easily get disorganized Highly customizable and thus causes confusion in larger teams Requires experts 	<ul style="list-style-type: none"> Outsourcing is hard due to high level of client interaction Managing many stakeholders is hard Difficult to apply in large projects with several stakeholders and team members 	<ul style="list-style-type: none"> Requires highly skilled people to execute on all four phases Multiple iterations add overhead

Figure 3.1 - comparison of development mythologies (Software development methodolgies comparison,2015)

After the comparison is being done, Spiral methodology was selected due to the following reasons

- High amount of risk analysis hence, avoidance of Risk is enhanced.
- Good for large and mission-critical projects.
- Strong approval and documentation control.
- Additional Functionality can be added at a later date.
- Software is produced early in the software life cycle.

(What is Spiral model- advantages, 2015)

3.4 Research Methodology

There are two categories of research methodologies such as inductive researching and deductive researching. In an inductive perspective to research, a researcher starts gathering data which is

necessary to his or her topic of interest. Deductive approach is aimed at proving and testing a proposal. Deductive research methodology was chosen for this project in order to perform testing and evaluation with the suggestions of the domain experts as mentioned in the introduction chapter. (Inductive or Deductive, 2015)

3.5 Chapter Summary

This chapter focused on project management which actually helped this proposed system to complete successfully. This chapter started with explaining about the project management methodologies where, PRINCE 2 was selected since it's widely used and easy to handle time management factor. Then the chapter moved on to talk about the time allocation which is one of the main task of the project. There after the chapter moved on to explain about the development methodologies where various kinds of development methodologies were compared and finally spiral development methodology was selected. Finally this phase discussed about the research methodology where deductive research methodology was selected among both inductive research and deductive research.

Chapter 4: System Architecture & Design

Contents

- Chapter Overview
- High Level Design
- System Design
- Design Goals
- Chapter summery

4.1 Chapter Overview

In this chapter mainly focused on the architecture and the design of the proposed IoT Security system. This chapter first discusses about the high level architecture and the selection of software methodology. Then the chapter talks about the system design with high level and low level diagrams. Finally this will discuss about the overall design goals for the proposed IoT Security system.

4.2 High Level Design

The high level design provides an accurate path and picture to the project process. In other words where it should start and where it should end and also the steps to implementation.

4.2.1 Rich Picture of the IoT Security System

A rich picture is so easy to explain what the system does and gives a better idea on how to implement it. It also gives a clear idea for the other developers where, if there's any implementation to be done in the future. Even the end user also can easily understand the system and the methodology.

This proposed IoT security platform mainly consist with following components.

1. Client side
2. Server side

4.2.1.1 Client side

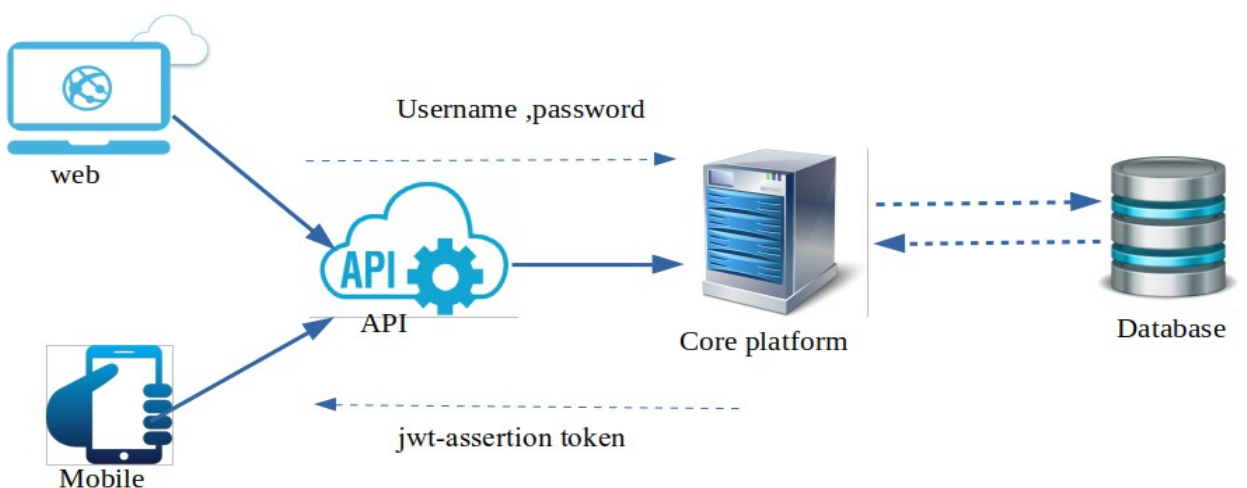


Figure 4.1 -Client side Rich Picture

according to above rich picture(Figure 3.1) the client architecture where,the user must get registered in the platform with the appropriate information such as (name,password, mobile no & etc) .

Client side validations will happen with the following steps:

- 1) Send username password to server side
- 2) Client server certificate validation happen
- 3) Username password validation(authentication) happen
- 4) If it is a valid user server will return jwt-assertion token

JWT-assertion token

“JSON Web Token (JWT) is a means of representing claims to be transferred between two parties. The claims in a JWT are encoded as a JSON object that is digitally signed using JSON Web Signature (JWS) and/or encrypted using JSON Web Encryption (JWE).In simple terms, it is just another way of encoding JSON object and use that encoded object as access tokens for authentication from the server.”(JWT to authenticate Servers API’s,2018)

JWT TOKEN



Figure 4.2 -JWT Token(JWT to authenticate Servers API’s,2018)

4.2.1.2 Server side

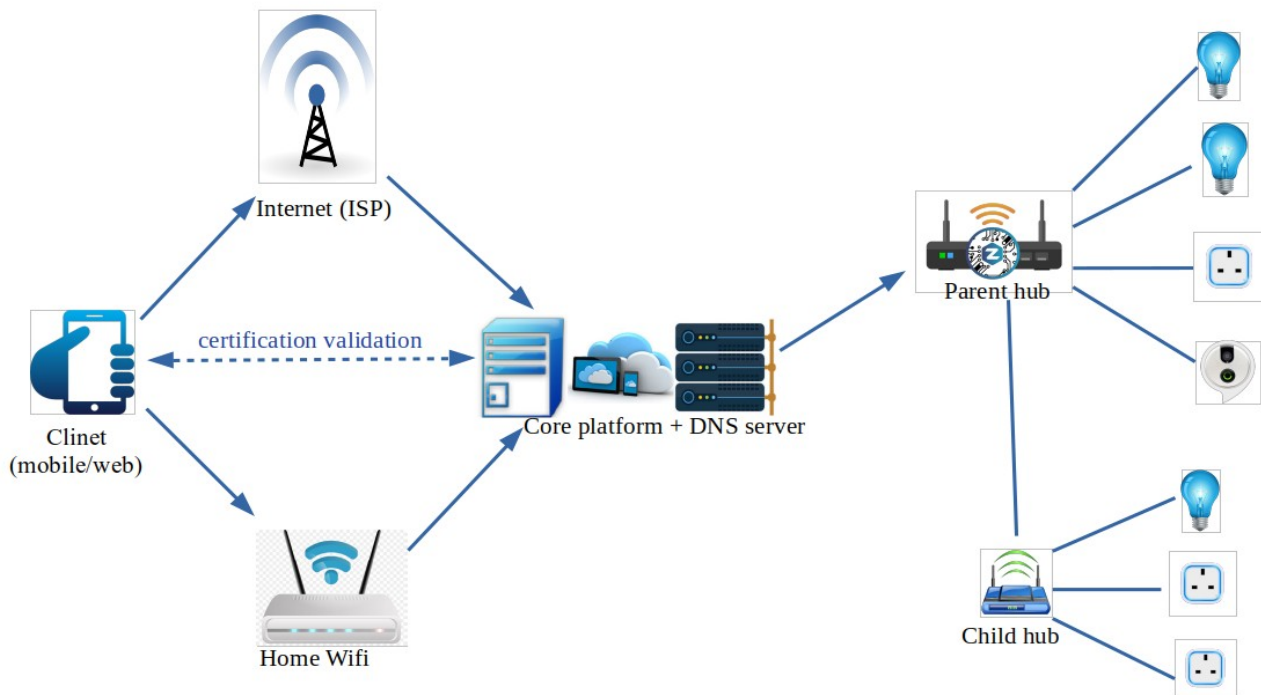


Figure 4.3- Server side Rich Picture

According to above rich picture (Figure 3.3) there is client is a mobile or web application which the user needs to use. This diagrams explains that if the user need to perform a action on IoT devices, which the user has access.

following steps are occur in server side when a user tries to perform an action

- 1) User send action to IoT device like DNS pattern (Eg:-blub12.Smart1blub.Room1,home)
- 2) Request ping local-host(direct connect to local-server) if get response from local-server then action sent through local-server else action perform though internet (this mechanism used because when user at same location no need to use internet and this mechanism can reduce delay)
- 3) validate client and server certificate and return jwt-assertion token
- 4) message go to local DNS server and find hub location and pass the message to hub(temporarily stored in database)
- 5) in hub active scanning will ping all devices with device id
- 6) hub get response from device which is related to message and with response hub get device path and pass message through that , else message time out send to client

7) return server response to client

4.2.2 High Level Architecture

The proposed IoT security system, needs to consider about the authentication, authorization and scalability. Therefore the author has used the three-tier architecture as the high level architecture of the proposed system.

4.2.3 Three-tier architecture

This is a client-server architecture in which the user interface, computer data storage, data access and functional process logic are implemented and preserved as independent components on different other platforms. This can be considered as a software design pattern. (What is Three-Tier Architecture? 2015)

4.2.4 High Level Architecture Diagram

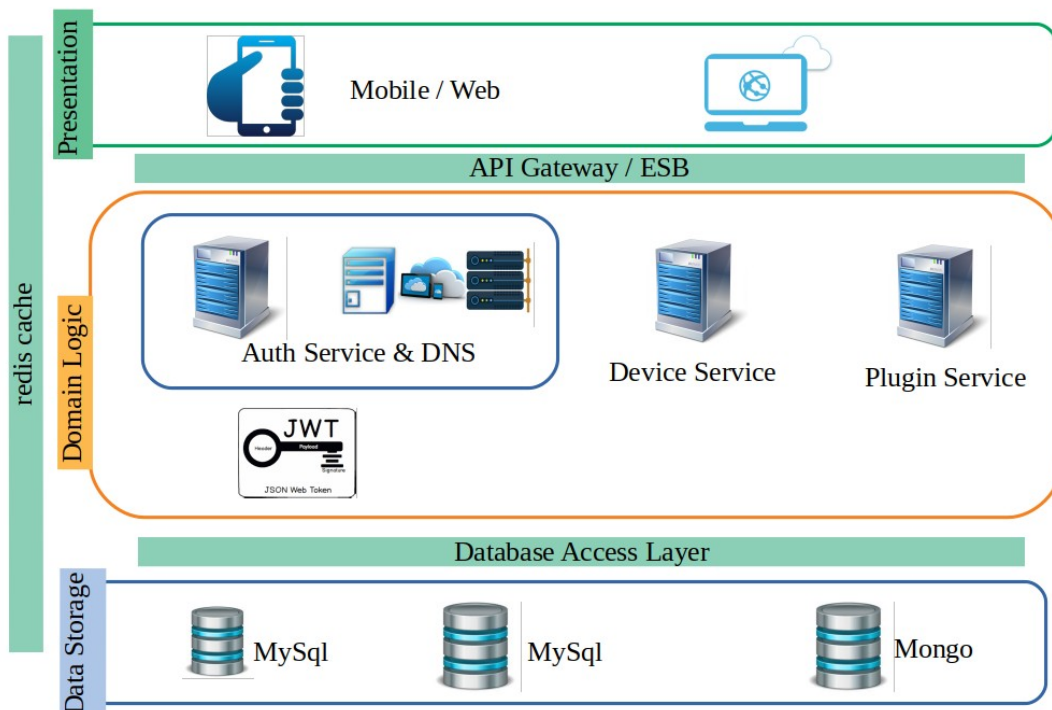


Figure 4.4 - High Level Architecture Diagram

According to the above High Level Architecture Diagram, presentation tier contains mobile or web client . This is the model which brings the proposed system to the outside world.

Domain logic tier contains the functionality of the system. According to the above diagram,It has Auth, Device, Plugin Services and DNS server to execute logic which are related to proposed system

Finally the data storage tier contains MySQL database and MongoDB.

MySQL Database: User details, Device details

MongoDB: key for HMAC

4.3 System Design

4.3.1 Selection of Design Methodology

Selection of the design methodology is important in order to development the proposed system successfully. The aim of this methodology is that, to develop the system with all the functionalities which were gathered from all the users within less time period. Due to the above reasons the author had to compare several methodologies which are available within the industry and choose one of them for the proposed system. Following are the methodologies which were compared.

- Structured Systems Analysis and Design (SSAD)
- Object-Oriented Analysis and Design (OOAD)

OOAD was chosen as the design methodology for the proposed system due to the following reasons:

- Easy to understand
This methodology has OOP concepts which is easy to understand and to translate complex theories into objects and class.
- Easy to Maintain
This methodology is widely used by the developers due to its maintainability. When compared to structured methodology this particular methodology is easy to operate.
- Data reuse
Use of OOP concept, which ideally mean that it's possible to use the data again.
- Re-usable

The OOP concept encapsulation enables the reuse of old data into new applications.

(Benefits Of Object Oriented Analysis and Design, 2015)

4.6 Design Goals for Overall Solution

This discusses about the Design Goals for the Overall Solution. Following are the main design goals for the overall solution.

Authentication

The proposed system's success mainly depends on the Authentication. Throughout the design chapter more analysis was given in order to gain the Authentication for the system.

Authorization

The proposed system's success mainly depends on the Authorization. Throughout the design chapter more analysis was given in order to gain the Authorization for the system.

Scalability

Scalability is another important part in proposed system's success. Where throughout that phase, deep review was given to gain the highest possible scalability for the proposed system.

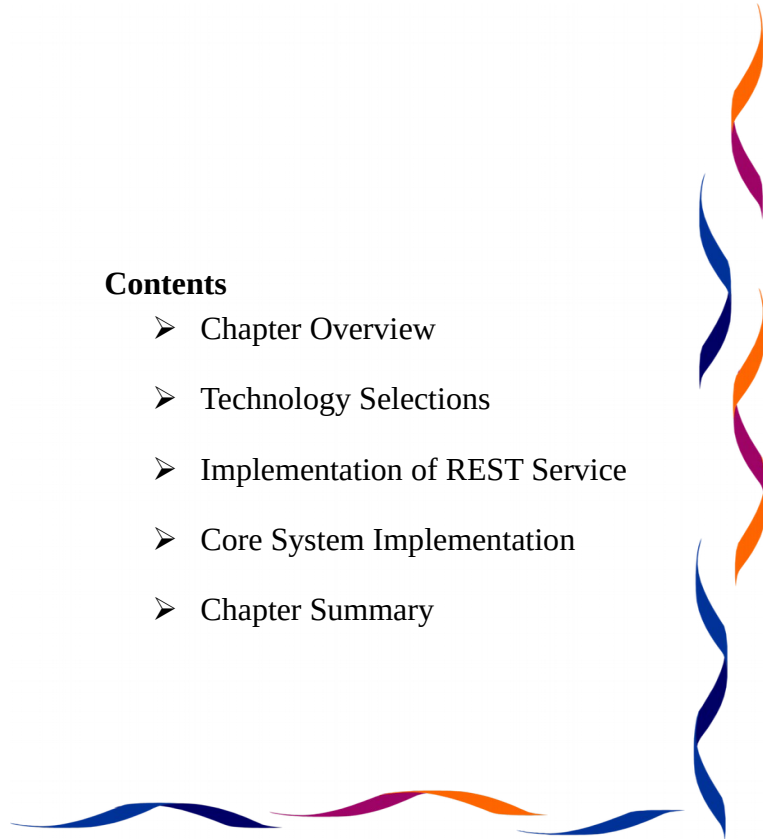
4.7 Chapter Summary

This chapter discussed about the proposed IoT security system architecture and the design. This chapter starts with the high level design, which discusses about Rich Picture of the IoT security System, high lever architecture and the high lever architecture diagram. Then the chapter moves on to discuss the system design where it discusses about the selection of software methodology and selection of language. Then the chapter moves on to the high level diagram. Finally this chapter discussed about the Design Goals for the proposed IoT security system such as accuracy, scalability and adaptability. The next chapter will be focusing on the Implementation of the IoT security system.

Chapter 5: Implementation

Contents

- Chapter Overview
- Technology Selections
- Implementation of REST Service
- Core System Implementation
- Chapter Summary



5.1 Chapter Overview

The previous chapter discusses about the design of the proposed system. This chapter is going to focus on implementation of the proposed IoT-Security system. This phase will be discussing about different kinds of technologies, frameworks, APIs and development environments which were used to implement the proposed system. This will be also discussing the problems which were occurred during the implementation with code snippets and screens shots.

5.2 Technology Selections

5.2.1 Selection of programming languages

This section discusses about which language is being used to develop the proposed IoT-Security system. The selection of the program language analyses various kind of languages. The previous chapter software architecture have discussed about which language is more preferable to implement this proposed IoT-Security system. Most of the research Software also have implemented with JAVA as the best language since this system mainly targets all kinds of platforms. Java supports and works fairly in minimum system requirements of a computer or a device.

5.2.2 Web service exposure approach

There are two kind of web services known as the SOAP which stands for Simple Object Access Protocol and REST which stands for Representational State Transfer. But the author is using the REST web services due to the reasons below.

- REST is faster than SOAP – there is no strict specification like SOAP and it works with less bandwidth and resource.
- Language and Platform independent- there is no such restriction like SOAP RESTful web services can be written in any programming language and executed in any platform.
- Can use SOAP- RESTful web services can use SOAP web services as the implementation.

- It is allow different data format- RESTful web service permits different data format such as Plain Text, HTML, XML and JSON.
- Security -Advanced security mechanisms adopted (RESTful Web Services,2015)

5.2.3 Selection of Framework for the web service

The proposed system's implementation has chosen spring-boot as the framework after comparing with many kinds of frameworks. Following are some of the reasons:

- Open source framework.
- exposing data in a variety of representation media types.
- abstract away the low-level details of the client-server communication.
- It is simple to develop.
- It has a portable JAX-RX API (Java programming language API).

5.2.4 Selection of an IDE and a deployment environment

There are various kinds of IDEs' available, in order to implement web services. Where most of the developers are using NetBeans, IntelliJ and eclipse. IntelliJ has been selected due to the following reasons such as open source, little bit faster than the others and easy to use. Since IntelliJ always checks the code when something is typed it and gives suggestions if there are any mistakes. It also takes less compiling time than other IDEs.

5.2.5 Selection of Third-party server

There are various kinds of cloud servers which are available to implement, such as Ubuntu cloud server, amazon EC2, Google Cloud server etc. When compared to all three servers the author has chosen Ubuntu cloud server due to the following reasons.

- single interface so easy to setup.
- in built security layer.
- works with low performance system.

- performs faster than the others.
- Role- base access.
- Open source.

5.3 Implementation of REST service

5.3.1 Implementation CURD API of user, device, location services

Core System Implementation

Core system is implemented with java language with the following code segments.

```
@RequestMapping(value = "/api/users/{id}", method = RequestMethod.PUT, consumes = MediaType.APPLICATION_JSON_VALUE, produces = MediaType.APPLICATION_JSON_VALUE)
public ResponseEntity<UserDTO> updateUser(@PathVariable("id") int id, @RequestBody UserInternalDTO userInputDto,
    HttpServletRequest request) throws IotException {
    setLogIdentifier(request);
    boolean accessResponse = apiAuthService.validateUser(request.getHeader(APP_HEADER_NAME),
        request.getHeader(USER_ID_BY_HEADER), request.getHeader(USER_SESSION_HEADER));
    if (accessResponse == false) {
        throw new IotException("Access Denied.");
    }
    User updatedUser = userService.update(id, modelMapper.map(userInputDto, User.class));

    if (updatedUser == null) {
        return new ResponseEntity<UserDTO>(HttpStatus.INTERNAL_SERVER_ERROR);
    }
    return new ResponseEntity<UserDTO>(modelMapper.map(updatedUser, UserDTO.class), HttpStatus.OK);
}
```

```
@RequestMapping(value = "/user/login", method = RequestMethod.POST, consumes = MediaType.APPLICATION_JSON_VALUE)
public ResponseEntity<?> userLogin(@RequestBody Map<String, String> params, HttpServletRequest request)
    throws IotException {
    setLogIdentifier(request);
    User user = userService.login(params.get("username"), params.get("password"));
    if (user != null) {
        UserDetailDTO userDetailDTO = modelMapper.map(user, UserDetailDTO.class);
        return new ResponseEntity<UserDetailDTO>(userDetailDTO, HttpStatus.OK);
    } else {
        return new ResponseEntity<String>( body: "Unauthorized", HttpStatus.UNAUTHORIZED);
    }
}
```

DNS resolver

```
package ucsc.mauran.iot.ims.service.impl;

import java.util.ArrayList;
import java.util.List;

public class DnsResolver<T> {
    private List<DnsResolver<T>> children = new ArrayList<DnsResolver<T>>();
    private DnsResolver<T> parent = null;
    private T data = null;

    public DnsResolver(T data) {
        this.data = data;
    }

    public DnsResolver(T data, DnsResolver<T> parent) {
        this.data = data;
        this.parent = parent;
    }

    public List<DnsResolver<T>> getChildren() {
        return children;
    }

    public void setParent(DnsResolver<T> parent) {
        parent.addChild(this);
        this.parent = parent;
    }

    public void addChild(T data) {
        DnsResolver<T> child = new DnsResolver<T>(data);
        child.setParent(this);
        this.children.add(child);
    }

    public void addChild(DnsResolver<T> child) {
        child.setParent(this);
        this.children.add(child);
    }
}
```

5.4 Chapter Summary

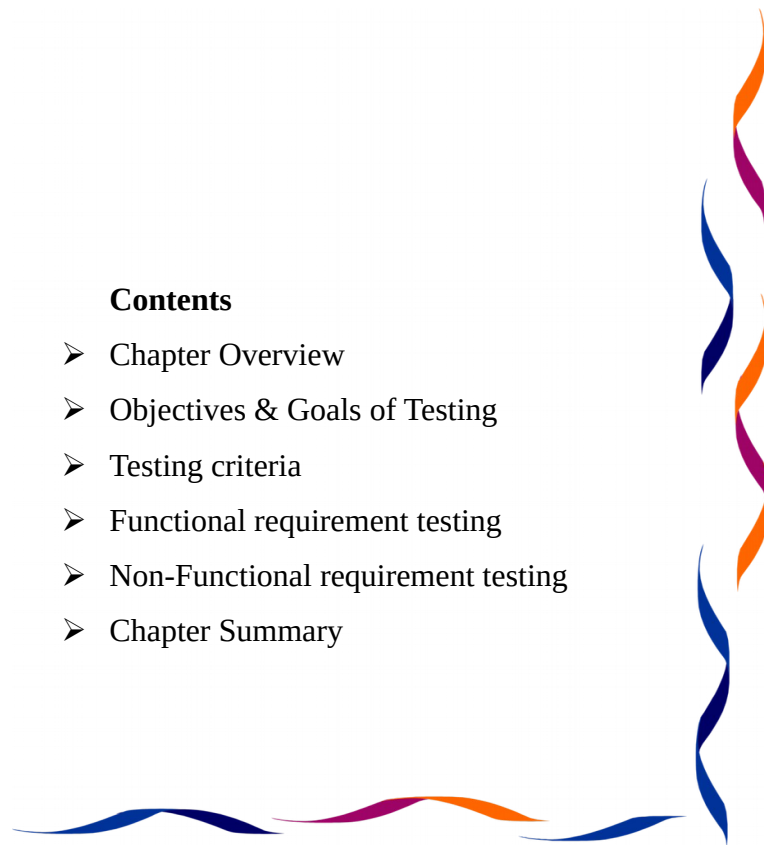
This chapter discussed about implementation of the proposed IoT-security system. This chapter starts with the selection of the technology and discusses about the selection of programming languages and the selection of IDE. Then the chapter moves in to the web services approach where it discusses about the web services and the spring-boot frame work which were used to implement this system. Then the chapter moves on to discuss about the Implementation of the REST service and the other core

components with appropriate code segments. Next chapter will be discussing about the testing details of the implemented system.

Chapter 6: Testing

Contents

- Chapter Overview
- Objectives & Goals of Testing
- Testing criteria
- Functional requirement testing
- Non-Functional requirement testing
- Chapter Summary



6.1 Chapter Overview

In the previous chapter, having discussed about proposed IoT-Security system Implementation, this chapter is mainly focus on testing the functional and non-functional requirements of the IoT-Security system. In this chapter will be discussing about the testing criteria , testing methods and testing levels. Finally this chapter will analyze the test results from various kinds of test methods.

6.2 Objectives and Goals of Testing

Software testing helps to confirm that the proposed IoT-Security system is working fine according to as we designed and implemented. This makes sure that the system's security, scalability, efficiency and other non-functional requirements are achieved. This chapter first starts with the testing criteria and then moves on to testing methods and testing levels where finally it discusses about the testing results. The major objectives of IoT-Security testing process are as follows:

- Finding errors and defects which may be created by the programmer while developing the software.
- To ensure proposed IoT-Security functional requirements are verified and validated.
- To ensure proposed IoT-Security non-functional requirements are verified and validated.
- To further enhance of the proposed IoT-Security system based on test results.

6.3 Testing Criteria

Testing the implemented proposed IoT-Security system is to ensure that its functional and non-functional requirements are working properly without any errors and bugs. Software quality can be measured in through software functional quality and Software structural quality.

1. Software functional quality

The software functional quality mainly focuses on the system function that match with the given technical requirement functions.

2. Software structural quality

The Software structural quality is used to measure the functional and non –functional requirements performance of the system.

6.4 White Box And Black Box Testing

This can be divided into two main categories such as white box testing and black box testing.

6.4.1 White Box Testing

White box is testing focuses on the coding structure and whether the codes are written properly. During this testing it inspects the code line by line and solves if there are any bugs or errors found.

6.4.2 Black Box Testing

Black box testing focuses on the output whether the system gives the expected output or not. This examines the system functionality without worrying about its internal structures or workings. This black box testing can be applied to every level of software testing such as unit testing, system, integration and etc.

6.5 Functional Requirements Testing

To ensure that the gathered requirements of the system are working correctly and the system gives the expected outputs for inputs. Testing method for functional requirement Functional requirement testing uses spiral methodology which is helpful to test the system with flexibility and freedom.

Testing method for functional requirement

Functional requirement testing uses spiral methodology which is helpful to test the system with flexibility and freedom.

No	Requirement description	Pass Rate	Status
01	new user should be able to register system with unique	100%	Pass

	username and password		
02	A registered user should be able to login to the system using username, encryption key and password	100%	Pass
03	A registered user should be able to get JWT- token when login to the system using username, encryption key and password	100%	Pass
04	System should authenticate the user	100%	Pass
05	System should give a notification for authentication and authentications system should display an error message to the user mentioning the reason for the authentication failure.	100%	Pass
06	User should able to add device into system	100%	Pass
07	User should able to update device which already in the system	100%	Pass
08	User should able to delete device which already in the system	100%	Pass
09	User able to add zone to system	100%	Pass
10	User should able to update zone which already in the system	100%	Pass
11	User should able to delete zone which already in the system	100%	Pass
12	User should able to search device and zone which already in the system	100%	Pass
13	System should display an error message when search fails	100%	Pass
11	System should display an error message when the search results are not found for search key word	100%	Pass

Table 6.1- Testing method for functional requirement

6.6 Non Functional Requirements Testing

6.6.1 Accuracy Testing

The proposed IoT-Security's main non-functional requirement is the accuracy of the resolving DNS. When a user needs to send a message to a device from the system it should give accurate device location for the key input which should be more than 98%. Here each and every functions are executed more than 15 times and the average time is taken by using the formula below.

$$\text{Accuracy} = \frac{\text{number of true positives} + \text{number of true negatives}}{\text{number of true positives} + \text{false positives} + \text{false negatives} + \text{true negatives}}$$

Following table (Table 6.2) shows the results and statistics of the accuracy testing process.

Test case	Purpose	Input data	Expected result	Actual result	Comment
1	Resolve a simple dns	unique_id.domain_name.	Resolve results true	true	Pass
2	Resolve a dns with action	unique_id.domain_name.action	Resolve results true	true	Pass
3	Resolve a dns with device id	unique_id.object_identifier. domain_name	Resolve results true	true	Pass
4	Resolve a dns with device id and subId	unique_id.object_identifier. subId.domain_name	Resolve results true	true	Pass
5	Action pass to correct to device	Action	Results true	true	Pass
6	Action pass to wrong to device	Action	Results false	false	Pass

Table 6.2- Accuracy Testing

According to above table all the accuracy testing has been a success. Above table shows that each and every search case's system response are correct and also according to the above accuracy equation 99% of accuracy was achieved for all the test cases. In this accuracy testing some test cases were omitted because those were kept aside as future enhancements. For more details read the conclusion chapter. Overall accuracy of the testing shows that all the test cases have given the expected results which means that the system gives accurate results for the user requests.

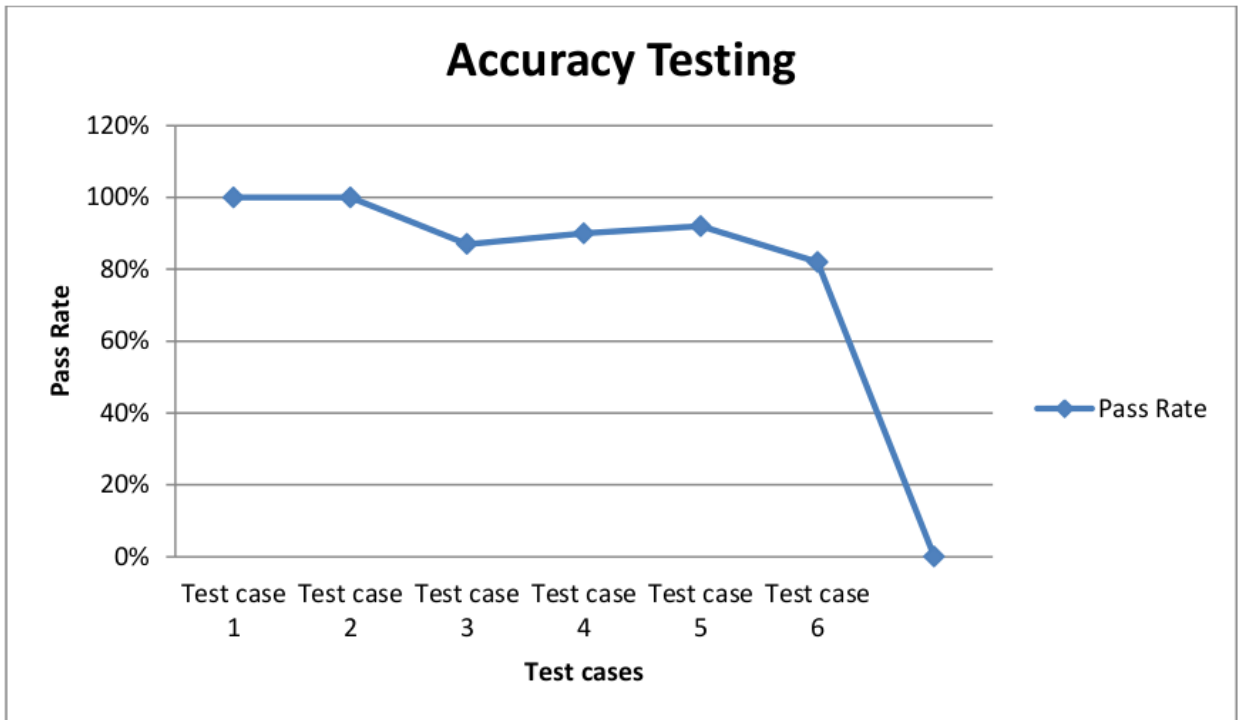


Figure 6.1- Accuracy Testing Chart

6.6.2 Performance Testing

The proposed IoT-Security’s system performance testing was carried out through the response time of each functions. Each and every functions were executed more than 15 times and the average time was taken using the following equation.

$$\text{Performance time} = \frac{\text{Total time for execution}}{\text{Number of execution}}$$

Following table (Table -6.3) shows the results and statistics of the performance testing process

Test case	Purpose	Input data	Expected result	Actual result (s)	Comment
1	Register to the sytem	Usarname, email,phone, etc	Less than 2 seconds	1.22	Pass
2	login to the system	Username and the password	Less than 2 seconds	1.12	Pass
3	Logout from the system	NA	Less than 2 seconds	0.56	Pass
4	Add a device	Some key word	Less than 3 seconds	3.16	Pass
5	resolving the dns	Zone, device and action	Less than 3 seconds	1.12	Pass

Table 6.3- Performance Testing Results

According to above table all performance testing have been succeeded. The above table shows that, when system executes the main functions which means resolving the dns, it only takes 1.12 seconds. These resolving function works less than a second which means that the system responses very quickly. Overall performance testing shows that all the test cases have given the expected results.

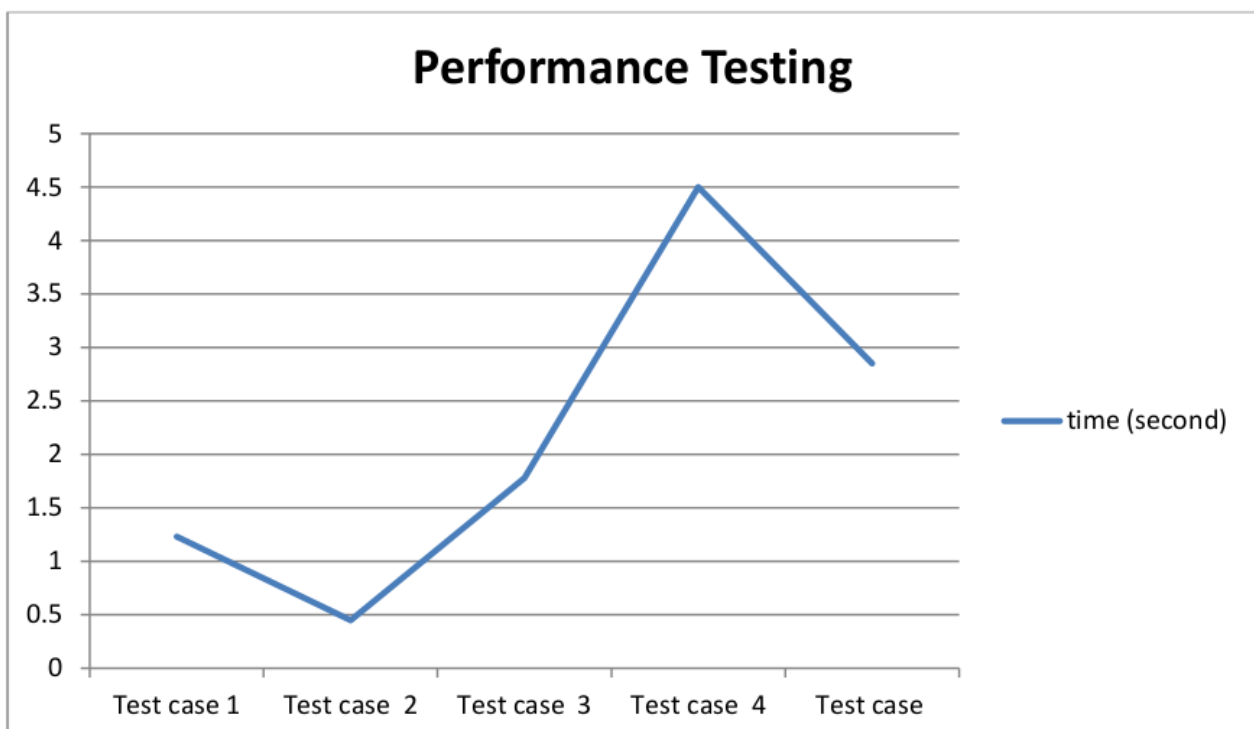


Figure 6.2 – Performance Testing Chart

6.6.3. Load and Scalability Testing

The proposed IoT-Security system Load and Scalability Testing was carried out to test the efficiency level of the system and also to test how the system would work with in the allocated resources. To do this kind of testing there are many tools available but the author has used Apache Benchmark and apache jmeter which are widely used for load and scalability testing. For this testing purpose, system was executed with the with Apache Benchmark tool in a machine with an i5 processor, 8GB RAM and Ubuntu 18.04 64bit operating system. Following results were derived after testing the system with the Apache Benchmark tool

Concurrency Level: 10

Time taken for tests: 10.045 seconds

Complete requests: 314

Failed requests: 3

Write errors: 1

Keep-Alive requests:113

Total transferred:297910 bytes

Total POSTed: 57680

HTML transferred: 86100 bytes

Requests per second: 2115.79 [#/sec] (mean)

Time per request: 10.453 [ms] (mean)

Time per request: 2.566 [ms] (mean, across all concurrent requests)

Transfer rate: 63.26 [Kbytes/sec] received

Summary of the above Apache Benchmark tool results is that the proposed IoT-Security has an acceptable rate for loading and scalability where the system also loads within a second. Connecting and disconnecting servers were done several times to confirm that the system works in a correct way and make sure that the system loads in the correct path.

6.7 Limitations of the Testing Process

- **Limited time period**

The proposed IoT-Security system should be tested more than the above test cases and also need to be monitored every month to identify how's the system working when data level is increased. But within a limited time period it's hard to do all kind of test cases.

- **Testing was carried in a local-host not on a real client server environment**

This testing was carried out from the same machine the client and server was running on the same network. Due to that current test result is not an accurate one since the system depends on the network speed and bandwidth. Sometimes the system performance will be low on real client server environment.

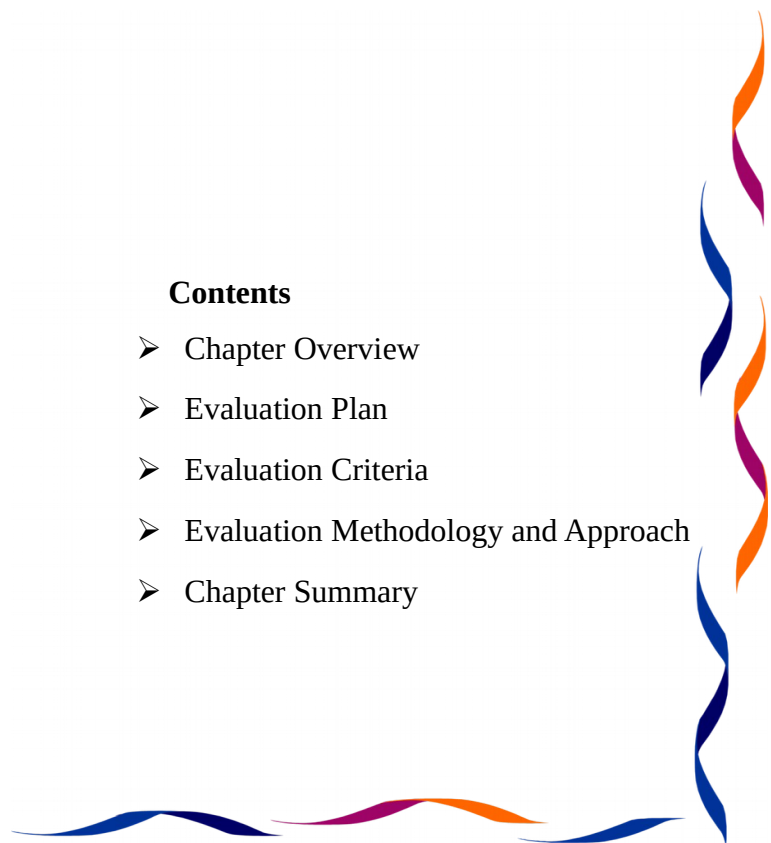
6.8 Chapter Summary

This chapter mainly focuses on the testing process of the proposed IoT-Security system. This chapter starts with the Objectives and Goals of Testing which discusses about why it's a must to test the system. Then the chapter moves on to discuss about the main types of testing such as Software functional quality testing and software structural quality testing. Then the chapter explains about white box testing and black testing. There after it talks about the functional requirements which uses spiral testing methodology because it helps to test the system easily and also test results has been shown with a table. Then the chapter moves on to non-functional requirements testing which focuses on testing the main non-functional requirements such as accuracy, performance and scalability which are shown in a table with the test results. Overall test cases was a success and the system worked with 100% accuracy level. Next chapter will discuss about the evaluation process which was carried out from various kind of evaluate methods.

Chapter 7: Evaluation

Contents

- Chapter Overview
- Evaluation Plan
- Evaluation Criteria
- Evaluation Methodology and Approach
- Chapter Summary



7.1 Chapter Overview

In the previous chapter the author has discussed about the proposed system’s testing methodologies such as functional and non-functional testing. This chapter focuses on the evaluation process carried with various kinds of methodologies and elevators. Finally it discusses about self-evolution of IoT security system with its Strengths and weakness.

7.2 Evaluation Plan

Services such as User management, Device management and plugin management host on remote servers (Amazon EC2) with Docker container

Client Application going to test on computer and mobile phone

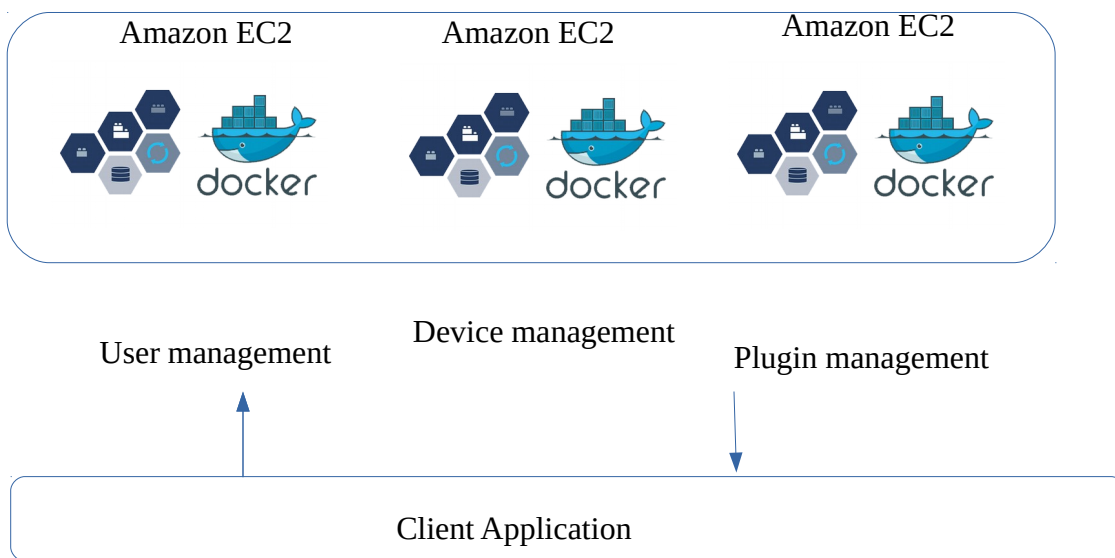


Figure 7.1 – Evaluation Plan

7.3 Evaluation Criteria

The following table shows the identified criteria for proposed project evaluation process. These criteria’ was mainly selected by covering the whole part of the proposed project.

Criteria	Description& purpose
----------	----------------------

The whole system Overall concept	In this criteria proposed system should obtain comments, views, feedback and evolution from all the stakeholders who are related in IoT field
the project Scope and depth	a good accurate IoT security has not been implemented on encrypted data algorithms and also cloud technologies area is broad so that it's important to find the scope and depth of the proposed system.
System design, architecture and implementation	To evaluate proposed system design, architecture and implementation of each methods.
The Solution and Prototype	To evaluate proposed solution and solve the identified IoT security problem.
The prototype usability, performance and accuracy of	To evaluate non-functional requirements of the proposed system in order to confirm, up to which level the non-functional requirements were implemented.
Limitations of the solution and future enhancements	To identify limitations of proposed system and future enhancements.

Table 7.1 – Evaluation Criteria

7.4 Evaluation Methodology and Approach

Every project success depends on evaluation process on users and domain experts' feedback. In this case for proposed system evaluation methodology gets feedback about major phases of the project life cycle such as problem, analysis, design and implementation. The system was evaluated by using less quantities and good qualities methods. The quantitative evaluation depends on testing phase of the system. This section mainly focuses on qualitative evaluation of the system. Evaluation of the proposed system was carried out through test by stakeholders, questionnaire and self-evaluation approaches, In these evaluation methodologies self-evaluation was mainly focused because of the time limitation of the system. For this evaluation various kinds of test plan were prepared. First test plan is about implemented protocol usage, accuracy, efficiently, security and other non- functional requirements also which is distributed to domain experts such as cloud engineers and security experts with the implementation source code design. Another test plan is prepared to measure architecture, design and implementation qualitative of the proposed system which was distributed among the

software engineers and software architects. The final test plan was prepared to measure the non-functional requirements' of both qualitative and quantitative of the proposed system which was distributed among the end users of the proposed system. The final evaluation process was carried out with the domain experts such as cloud engineers and security experts to evaluate quality, concept, scope and prototype.

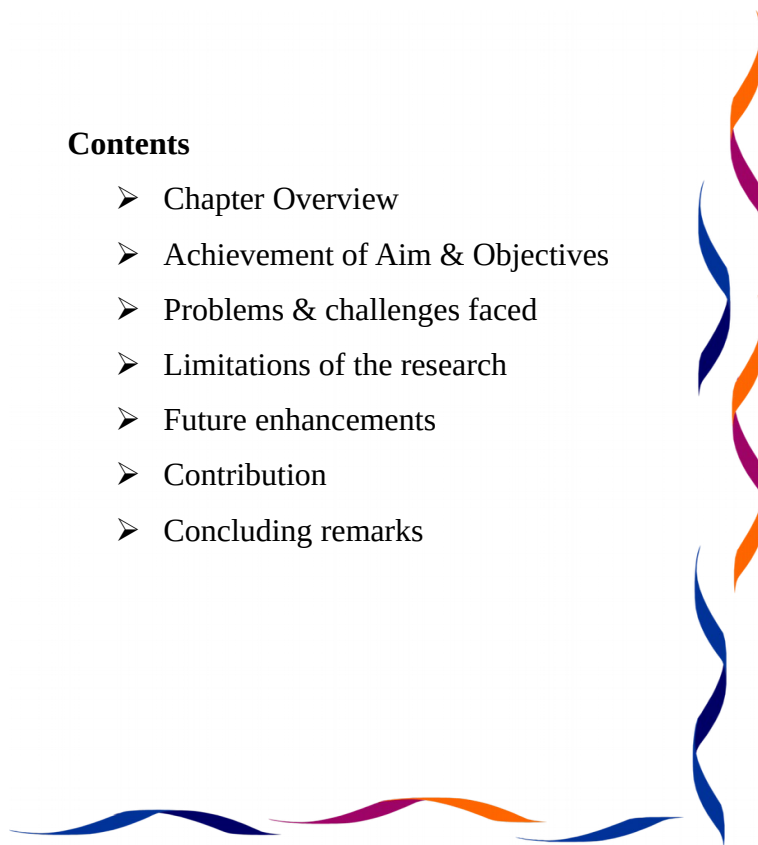
7.5 Chapter Summary

This chapter focused on system evaluation. In this chapter evaluation was discussed first which is about criteria such as evaluation methodology and various kind of evaluators. Then the chapter moved to Justification which is about approaches that were selected in questionnaire and interviews. Most of the stake holders gave positive feedback about the system idea and it has been reflected in the project success. Then the chapter focused on project scope and depth. Most of the evaluators mentioned that the IoT security system design goals and the selection of technology for the implementation was good enough for system. This also focused on non-functional requirement of the proposed IoT security system system and got positive feedback about non-functional requirement from evaluators. As an overall, in this chapter most of the evaluators have given positive feedback about the system. Finally this chapter discusses about self-evaluation. Next chapter going to discuss about the conclusion of the project.

Chapter 8: Conclusion

Contents

- Chapter Overview
- Achievement of Aim & Objectives
- Problems & challenges faced
- Limitations of the research
- Future enhancements
- Contribution
- Concluding remarks



8.1 Chapter Overview

The previous chapter discusses about the evaluation process with different kinds of evaluation methods about the proposed IoT security system. In this chapter, going to focus on the conclusion of the proposed IoT security system with highlighting the achievements of the aim and objectives and also going to discuss about the project life cycle with problems and challenges faced during implementation, limitations of the system, future enhancement and authors' project overview.

8.2 Achievement of Aim and Objectives

8.2.1 Aim

To design, develop and evaluate a protocol for Internet of Things; To pass messages and actions through IoT devices, with authentication, authorization and scalability. Further elaborating the project's aim, there are some existing solutions for this problem. But most of them are not accurate with current internet technology (Corser et al., 2017). Therefore, through this protocol, the user will be able to pass messages to IoT devices with authentication, authorization and scalability. The proposed system would support devices which are connected between them or connected to internet with a minor configuration. The proposed system will not cover security layers of third party servers, third party servers encryption & decryption algorithms.

8.2.2 Objectives

Project Objectives

- IoT and IoT security - to understand how IoT and IoT security works.
- Existing IoT security methodologies - to understand how existing methodologies work and analyze their techniques along with their strengths and weaknesses
- Compare with the existing IoT security methodologies - which were used by other technical experts to provide security for IoT.
- Through this proposed protocol, the user will be able to pass messages to IoT devices without losing any secrecy and confidentiality.

Other Related Objectives

Objective 1 IoT and IoT security - to understand how IoT and IoT security works.

Preparing the Terms of Reference contains:

- Chapter overview
- Previous work done
- Project background
- The aims of project objectives
- Features of the prototype
- Resource requirements
- Activity schedule.

TOR is included as chapter 1 of the report.

Objective 2 Existing IoT security methodologies

Carryout an in-depth literature survey

- Learn if there's any existing solutions regarding the issue and how can the project differ from the existing ones.
- discuss about, why does it need to protect IoT messages which also discusses along with critical evaluation. Finally it would analyses what tech people have already done for it along with related work done on the problem domain.

The Literature Review is included as chapter 2 of this report.

Objective 3 Compare with the existing IoT security methodologies

- Learn if there's any existing solutions regarding the issue and how can the project differ from the existing ones.

The Literature Review is included as chapter 2 of this report.

Objective 4 Through this proposed protocol the user will be able to pass messages to IoT devices without losing any Secrecy and confidentiality

- Discuss about, why does it need to protect IoT messages which also discusses along with critical evaluation.

Objective 5 Prepare a software requirement specification (SRS)

Carryout an in-depth Requirement gathering process

- Gathering requirement

- Gathering requirement about proposed security system and algorithm from domain experts such as software engineers ,software architectures and security experts and analysis there requirement
- Evolution about gathering requirement and identifying important requirement.

Objective 6 Determine the tools and technology (software and hardware) to be used in the development.

- Select the most appropriate technologies, tools (IDE), APIs, libraries, platforms, algorithms and hardware requirements to implement the proposed project prototype

Objective 7 Develop the project (prototype)

- Develop project prototype based on requirement gathered from related field works

Objective 8 Testing of the project(prototype)

- Testing Project appropriate manner such as creating test cases and testing the functional and non-functional requirements of the implanted IoT security system. In this and also discuss about testing criteria , testing methods and testing levels.

The Testing included as chapter 6 of this report.

Objective 9 Carryout a critical evaluation

the evaluation process carried with various kind of mythology . And also going to discuss about self-evolution of IoT security system with its Strengths and weakness.

The evaluation included as chapter 7 of this report.

Objective 10 Carryout a review from the experts

Had a discuss with varies kind of domain experts about proposed solution and prototype and evaluate and document it .The evaluation included as chapter 7 of this report.

Objective 11 Modification to the prototype

If there are any changes to be done with the prototype, should be changed accordingly.

Objective 12 Documentation

Document all the findings and the key steps involved in the project.

8.3 Problems and Challenges Faced

The depth of the scope

The IoT security protocol scope is earlier thought was scope is small so can do within a allocate time period but after depth research only known about it is bigger then expected so need to narrow down the scope with literature review and system requirement process.

Lack of knowledge and learning resources

The technologies which were used in implementation such as cloud, REST web services, spring frame work and IoT technologies knowledge were average on starting on this project so face many kind of implementation issues but solve them with self-learned from online documents. And also faced a problem to combine libraries and languages during the implementing process.

Limited Time Period

The proposed solution depth and scope is too big for allocated time period. And also lack of knowledge is increase time period then expected because of self-learned presses and also testing the prototype took more time period to successfully complete this system.

8.4 Limitations of the Research

Knowledge of existing solutions

This proposed IoT security mechanism unable to compare with existing solutions because some solutions haven't published well with that advantages and drawbacks and also most of research papers didn't mention tool which they use to implement that algorithms so it is hard to studying properly existing solutions

Restriction on the key success factors of a proposed IoT security system

In this research mainly forced on authentication, authorization and scalability as key success factor But there are many other key success factor apart from authentication, authorization and scalability

in limited time period unable to focused on other key success factor and this research can be further expanded to explore the other key success factors of proposed IoT security system.

8.6 Future Enhancements

Due to limited time period , lack of knowledge and project depth and scope some components related to the system didn't implement at this level and also it is possible to some other parts to the system. Following table showing future enhancement of the system.

Enhancement User have option to search device from index

Priority Level Medium

Lack of time unable to implement search device from index option , This additional option used to save users time because it will function like this if users type a search device it will checks index that word previously searched or not if yes reveal the search results through index or else do search with alternative method

Enhancement Integrate current system with security layer

Priority Level Medium

There are some possible way to integrate this system to current security layers which are using by IoT cloud services. Due to security reasons unable to implement it this period.

Enhancement Increase efficiency of DNS algorithms

Priority Level High

There are some minor problem in algorithm lack of time period unable to solve it. It will help increase efficiency of algorithms

Enhancement Increase performance and accuracy then current level

Priority Level Low

There are some possibilities to increase accuracy level of search results such as running separate client side and server side and changing the better algorithm then this.

Enhancement Integrate current system authentication with cloud authentication

Priority Level Medium

The current system has separate authentication system then cloud authentication. Better in future add both of them as one authentication such as Oauth2.

8.7 Contribution

This proposed IoT security system aim is provide a solution for current security problem in third-party servers to protect IoT messages and data. For this research almost evaluate every existing related to

proposed system. The developed IoT security system have solve the security problem of tried party servers through the IoT messages and data. Limitation faced on when comparing the DNS algorithm with excisting algorithms and also faced another limitation that managing the scalability of the system with the increasing amount of data. The IoT security system has contributed the users to secure their personal IoT messages and device data in third-party servers by providing own algorithm to search on encrypted data.

8.8 Concluding Remarks

The proposed IoT security system is a solution for protect messages through IoT devices. This system supports different type of devices with different platforms and low bandwidth. The above presented solution can be vastly beneficial for third party devices and IoT server users because of its accuracy, scalability and performance. The current solution opens up new research areas for the research community to further enhance the solution by trying out the future enhancements mentioned above.

References

- Tianbo Lu ,Xiaobo Guo ; Bing Xu ; Lingling Zhao ; Yong Peng ; Hongyu Yang, 2013. Next Big Thing in Big Data: The Security of the ICT Supply Chain. Alexandria, VA, IEEE, p. 8.
- Defining IoT Business Models, 2017 Monetising IoT investments, maximising IoT skills and addressing IoT security,Canonical Ubuntu
- Somayya Madakam ; Hema DateSpringer,2016.IT Applications Group, National Institute of Industrial Engineering (NITIE), Mumbai, India
- Wei Zhou ,Yuqing Zhang, Peng Liu ,2018 with the National Computer Network IntrusionProtection Center, University of Chinese Academy of Sciences, Beijing 100000,China,IEEE
- Jyoti Deogirikar ,Amarsinh Vidhate ,2017. Security Attacks inIoT: A SurveyInternational conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC 2017)
- Canonical. 2018. The leading operating system for PCs, IoT devices, servers and the cloud | Ubuntu. [ONLINE] Available at:
https://pages.ubuntu.com/rs/066-EOV-335/images/Canonical_Defining%20IoT%20Business%20Models_Final.pdf. [Accessed 06 June 2018].
- Almira Hamzic Isabel Olofsson. 2016. DNS and the Internet of Things. [ONLINE] Available at: <http://www.diva-portal.org/smash/get/diva2:1080720/FULLTEXT01.pdf>. [Accessed 5 June 2018].
- Baseline Security Recommendations for IoT — ENISA. 2018. Baseline Security Recommendations for IoT — ENISA. [ONLINE] Available at:
<https://www.enisa.europa.eu/publications/baseline-security-recommendations-for-iot>. [Accessed 07 June2018]

- IoT Agenda. 2018. *What is internet of things (IoT)? - Definition from WhatIs.com*. [ONLINE] Available at: <https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT>. [Accessed 08 June 2018].
- George Corser, Glenn A, Mohammed Aledhari, Jared Bielby, Rajesh Nighot, Sukanya Mandal, Nagender Aneja, Chris Hrivnak, Lucian Cristache
INTERNET OF THINGS (IOT) SECURITY BEST PRACTICES ,2018. [ONLINE] Available at: https://internetinitiative.ieee.org/images/files/resources/white_papers/internet_of_things_may_2017.pdf. [Accessed 08 June 2018].
- Software development methodolgies comparison. 2015. Software development methodolgies comparison. [ONLINE] Available at: <http://www.slideshare.net/anirudhoswal/software-development-methodolgies-comparison-v1-ao-exout> [Accessed 25 September 2018].
- What is Spiral model- advantages, disadvantages and when to use it?. 2015. What is Spiral model- advantages, disadvantages and when to use it?. [ONLINE] Available at: <http://istqbexamcertification.com/what-is-spiral-model-advantages-disadvantages-and-when-to-use-it/> [Accessed 25 September 2018].
- Inductive or Deductive? Two Different Approaches. 2015. Inductive or Deductive? Two Different Approaches. [ONLINE] Available at: <http://2012books.lardbucket.org/books/sociological-inquiry-principles-qualitative-and-quantitative-methods/s05-03-inductive-or-deductive-two-dif.html> [Accessed 25 September 2018].
- IoT Agenda. 2018. *What is internet of things (IoT)? - Definition from WhatIs.com*. [ONLINE] Available at: <https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT>. [Accessed 25 October 2018].

- Carritech Telecommunications. 2018. *Internet of Things: Explained | News | Carritech Telecommunications*. [ONLINE] Available at: <http://www.carritech.com/news/internet-of-things/>. [Accessed 27 October 2018].
- SharpNode Blog | Latest in IoT. 2018. *Benefits of IoT | SharpNode Blog | Latest in IoT*. [ONLINE] Available at: <https://sharpnode.com/post/benefits-of-iot-at-home/>. [Accessed 27 October 2018].
- codeburst. 2018. JWT to authenticate Servers API's – codeburst. [ONLINE] Available at: <https://codeburst.io/jwt-to-authenticate-servers-apis-c6e179aa8c4e>. [Accessed 25 November 2018].
- What is Three-Tier Architecture? - Definition from Techopedia . 2015. What is Three-Tier Architecture? - Definition from Techopedia . [ONLINE] Available at: <http://www.techopedia.com/definition/24649/three-tier-architecture> [Accessed 25 November 2018].
- Benefits Of Object Oriented Analysis and Design | Benefits Of. 2015. Benefits Of Object Oriented Analysis and Design | Benefits Of. [ONLINE] Available at: <http://benefitof.net/benefits-of-object-oriented-analysis-and-design/> [Accessed 26 November 2018].
- Security Intelligence. 2019. A Primer on IoT Security Risks. [ONLINE] Available at: <https://securityintelligence.com/a-primer-on-iot-security-risks/>. [Accessed 27 February 2019].
- GAO Assesses IoT Vulnerabilities - BankInfoSecurity. 2019. GAO Assesses IoT Vulnerabilities - BankInfoSecurity. [ONLINE] Available at: <https://www.bankinfosecurity.com/gao-assesses-iot-cybersecurity-other-risks-a-9926>. [Accessed 26 February 2019].