A STUDY ON THE FACTORS AFFECTING THE ADOPTION OF IOT SYSTEM IN A DEVOPS ENABLED ENVIRONMENT

By

BHAWANI SHANKAR MAHAWAR

THESIS

Presented to the Swiss School of Business and Management Geneva

In Partial Fulfillment

of the Requirements

for the Degree

GLOBAL DOCTOR OF BUSINESS ADMINISTRATION

SWISS SCHOOL OF BUSINESS AND MANAGEMENT GENEVA

JUNE 2024

A STUDY ON THE FACTORS AFFECTING THE ADOPTION OF IOT SYSTEM IN A DEVOPS ENABLED ENVIRONMENT

By

BHAWANI SHANKAR MAHAWAR

Supervised by

Prof. Kishore Kunal

APPROVED BY

_____

Vasiliki Grougiou

Dissertation chair

RECEIVED/APPROVED BY:

_____

Admissions Director

DECLARATION

I hereby affirm that the thesis named "**A Study on The Factors affecting the adoption of IoT System in a DevOps enabled Environment**" submitted to SSBM, Geneva for the award of the Doctor of Business Administration degree, is the result of my own study. This thesis, or any portion of it, has not been submitted, either partially or in its entirety, to fulfill the requirements of any degree program at any other university or institution.

(Bhawani Shankar Mahawar)

ACKNOWLEDGEMENTS

I would want to extend my utmost admiration, gratitude and thanks to my mentor Dr Kishore Kunal who has been a great source of inspiration, support, and advice during this research endeavor. His extensive knowledge, patience, and unwavering commitment have been instrumental in molding this thesis.

I express my gratitude to SSBM University and its management for providing me with the opportunity to undertake my doctoral studies.

I want to dedicate this work to the memory of my late father, Parameswar Dayal Mahawar. His wisdom, encouragement, and belief in my abilities continue to inspire me every day. Though he is no longer with us, his influence and guidance remain a guiding light in my life and academic pursuits.

Finally, I thank my mother, Kamala Devi Mahawar; my wife, Monalisa; and my brothers, Lakhmikant and Pawan Kumar, and my sister, Sabita, and my friends, Shivam and Ramprakash, for their prayers and moral support, which greatly significantly facilitated my ability to successfully carry out this research. I owe everything to my family; their support, and encouragement have fueled my academic journey and empowered me to overcome any obstacle.

(Bhawani Shankar Mahawar)

ABSTRACT

A STUDY ON THE FACTORS AFFECTING THE ADOPTION OF IOT SYSTEM IN A DEVOPS ENABLED ENVIRONMENT

Purpose

DevOps is, without a doubt, universally understood for its ability to accelerate delivery and enhance reliability in applications. Numerous stories of success in improving customer service have been pegged on the idea of DevOps. However, DevOps is still not widely adopted in the IoT field, where many components such as hardware, software and firmware are involved. The objective of this research is to explore the factors influencing the adoption of Internet of Things (IoT) systems within DevOps-enabled environments. The objective of this study is to identify and comprehend the various challenges and opportunities organizations face when integrating IoT technologies with their existing operational and development frameworks.

Design /Methodology

The study is based on original data collected from a targeted sample of 450 participants.. A conceptual framework incorporating UTAUT and the TTAT model been established. The study utilized purposive sampling for study, and data was collected through a carefully designed and a questionnaire that has been evaluated in advance. The study employed PLS-SEM analysis to evaluate the theoretical model.

Findings

The findings of the thesis indicate that the primary factor impacting the adoption of IoT systems in DevOps environments are perceived threat and effort expectancy. In contrast, the factors of perceived severity and perceived susceptibility yield a significant influence on the intention to use through perceived threat. Although there are other aspects like facilitating condition,

performance expectancy, and social influence also play roles in adoption, their impact is relatively lower.

Research Limitations

Although this study has identified security as a critical component of perceived threat, there still exist potential limitations that future studies can address. More to the point, future research should further identify the complexities regarding the aspects of security characteristic of IoT systems. As such, novel frameworks should be developed to try and analyze new areas more fully and exhaustively. Moreover, regulatory measures should be taken accordingly, and the new approach to risk management characteristic of DevOps should be adapted to fit the various conditions set to difference the deployment of IoT. Therefore, addressing in more detail this approach, researchers will be able to make a more substantial contribution in terms of improving the security of system, as a result, streamlining the process of deployment of IoT within DevOps workflows.

Originality

Despite the existence of very few studies on IoT adoption or DevOps practices, none have explored the two topics together. This paper, therefore, combines the two domains into one paper, hence contributing to existing literature. In addition, using the Importance-Performance Map Analysis method and models such as the UTAUT and TTAT to analyze the adoption dynamics brings a new aspect to the existing literature. Therefore, based on the novel context, methodology, and analytical frameworks, the research can be seen as original and of good depth.

Furthermore, the inclusion of case studies provides real-world examples to support the theoretical framework and considering the impact of leadership and social dynamics on IoT adoption, thus enhances the practical relevance and applicability of the research findings. The

outcomes and findings can help future organizations adopt IoT in a DevOps-enabled environment.

KEYWORDS

IoT, DevOps, Security, UTAUT, TTAT, Social Influence, Perceived Threat, Intention to Use, PLS-SEM, Hardware, firmware.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1 : INTRODUCTION

1.1 Introduction to Topic

Earlier people used to communicate via internet by exchanging mail, chat etc. Internet was mostly used to connect people around the world, and it was termed as internet of people. With progress of technology, we are now able to connect things\ smart device to internet and with each other and now we call it as (IoT)Internet of Things.

Internet has touched every aspect of our life with new smart IoT devices that are coming to market now a days. All personal data starting from our day-to-day activity like sleep routine, emails, messages, personal conversation, even the number of times our heart beats per minute are on internet. Industries are equipped with thousands and lakhs of devices and sensors that generate millions of confidential data for a company and send it to cloud. However, maintaining these devices, managing trust in this communication and promising secure data transfer is a big challenge which traditional security architecture or deployment cannot handle. **Alex Koohang(2021),** stated that use of IoT device is increased to 50.1 billion from 8.7 billion in last eight year. **M. Asad(2020)** clearly said that the increased usage of IoT devices\sensors in the organization brings with it deep concern over threats, attacks, and exploits.

(Pereira et al., 2021) Mentioned that Development and operation teams are driven by different goals and purposes. Development teams concentrate on delivering fast service to the business. On the other hand, operation teams try to ensure a stable, reliable, and secure service to customers. To deal with these differences, organization embrace a collaborative culture known as DevOps, which involves both teams working together. DevOps enables firms to integrate their activities, leading to a higher frequency of deployments and a more stable production environment. Consistently, Internet of Things (IoT) solutions face the same problems due to the required alteration of the customary building and maintenance methods. IoT solutions are made up of objects which are connected to one another, as well as to users, to collect and

analyze data that will serve specific objectives. They have multiple software components such as front and back ends of clients and servers. They also contain essential software for hardware including firmware, gateway, and edge computing.

(López-Peña et al., 2020) Today we have many different types of IoT applications, which use in industry, healthcare, and transportation. They are commonly found in places such as cities, industries, and hospitals. A number of these applications are systems which are supplied with sensors, computers, and other things to control the environment or monitor important things.

These IoT systems have two main parts: physical things like hardware and networks, and an application software that helps it function. They also possess several apps that do a particular job. Maybe, some of these systems provide very significant services, like those in healthcare or energy. However, since they're complex, it's not easy to supervise them closely so that they don't slip out of control. Also, Ensuring that they are always updated and function well as a team requires teamwork from different teams.

To tackle this problem, there is something called DevOps which is all about getting the different IT teams to work more smoothly together. It allows for shorted iterations of fixing and updating with the help of fast feedback between the teams.

In automation, DevOps stands out as a significant technology. Its primary objective is to accelerate the development process to ensure fast product and service delivery. Even though we have entirely used DevOps for unit testing and automated testing processes, penetration testing in the context of DevOps is not much discussed.

(Hiremath ,2023) With the increase in DevOps and agile models, the penetration testing has evolved from being done at the end of the development cycle to being done as development continues throughout the entire. With the use of DevOps and agile, we shall think out security since it is an important part of the software development process. DevSecOps is a method of

integrating security at every point and ensuring it is part of the programming, testing, and deploying processes. This guarantees that security is always given the precedence over other considerations, with constantly secured team participating closely with the developers. Organizations can be proactive in addressing security issues by acquiring DevSecOps and Penetration testing products. They do this by employing advanced technologies and techniques to detect and fix vulnerabilities. Such an approach can reduce the attack success rate.

With increase in number of smart devices the security threat for IoT is increasing day by day. The IoT Device or low embedded device are more prone to security attack which don't go through proper security testing and process. Attackers find attack surface on vulnerable device and try to hack the device to either steal personal data, jeopardize the device, or assassinate someone like for example what if you have an enemy, and he hacks your pacemaker device and make it stop your heart.

Franklin (2020) Observed that numerous small enterprises hastily and confidently introduce new products to the market in order to generate substantial profits. A significant number of individuals have difficulties and are unable to derive financial benefits from their IoT innovation, primarily due to security and maintenance issues. Security risks in IoT development and prototyping are frequently underestimated due to a lack of adherence to proper protocols.

As per **(Veluvarthi, 2023)** IoT is the fastest growing industry that will grow from $300.3 billion to $650.5 in 2021 to 2026 respectively. The matter of how we can protect them from dangers and how to ensure our privacy is one of the most important themes. By smart thermostats, security cameras and other IoT devices however not only ease life but also can be attacked by hackers if not properly deployed and maintained.

(Cheruvu, Kumar, Smith, Wheeler, 2020) State that Security of IoT devices, being a complicated process, involves many steps (parties). At the beginning of the process, building and protecting your online environment using security software is a matter of utmost importance. Next, in both the selecting and altering stage, the tools will need to be able to talk to different CPUs. The deployment should ensure applications run in such a way that they are accessible and might be used even anonymously, respecting security rules. Rather than that, they should consider the process of secure and efficient maintenance of these devices from retiring to recycling.

1.2 Research Problem

Till now we have seen that how DevOps has revolutionized the software industry by providing faster software delivery, collaboration stability through continuous feedback and automation. Research in the current adoption of IoT in the DevOps environments is minimal but in growth. Many research have discussed IoT adoption in various areas such as industry, smart homes and other sectors which DevOps practices in software development were also studied but there is a gap in research on the integration of IoT systems with DevOps environments.

Even though both IoT and DevOps have become more and more popular, there is an evident gap among the literature of the multiple factors that impact the adoption of IoT systems in the DevOps environments. A lot of the previous studies do not consider the special challenges and market conditions that arise due to this integration, because of this there is still not a complete understanding of the most important aspects that organizations should consider if they want to implement IoT systems in DevOps environments.

The void in IoT adoption factors within DevOps environments is a substantial challenge. It fails organizations to fully use IoT technologies leading to inefficiencies and a failure to innovate opportunities. Closing this gap is critical for enhancing organization’s productivity, promoting innovation, and guaranteeing safety in the volatile world of IoT and DevOps

integration. Thus, there is a need to study the various factor that affects the adoption of IoT System in a DevOps enabled environment.

1.3 Purpose of this Research

The main goal of this research is to identify the determinants of IoT systems adoption in DevOps-enabled environment. The main objective of this study is to determine and comprehend the different types of challenges and opportunities that arise in the process of implementing IoT technologies in the operational and developmental frameworks of organizations. Encompassing the ease of integration, the expected performance improvements, the impact of organizational culture and how much effort to adopt such technologies, technical and operational readiness of an organization for IoT adoption is the aim of the research.

Furthermore, this research will analyze the security threats of IoT adoption, studying how perceived threats and vulnerabilities influence the decision-making process. These findings are critical in understanding the obstacles that currently prevent the widespread adoption and in identifying the necessary measures for mitigating the possible risks.

The study seeks to carry out an investigation on how the IoT can either complicate or improve the continuous integration and deployment pipelines that are typical of DevOps practices. It will as well present practical measures and good practices on successful integration of IoT systems into these environments which will lead to operational efficiencies and innovations. The results are anticipated to arm IT managers and implementers with insights into the intricacies of IoT technology, allowing them to make decisions and plan strategically for a successful implementation.

To satisfy the study's purpose, the following objectives were framed:

1. To examine the factors that influence the implementation of IoT systems in DevOps-enabled environments.

2. To identify the challenges hindering the integration of IoT technologies into DevOps practices.
3. To assess the impact of these motivating factors and challenges on the effective use of IoT devices within DevOps workflows.
4. Identify primary security issues related to the integration of IoT systems into DevOps environments and understand how organizational stakeholders recognize these issues.
5. Analyze the impact of perceived risks on decision-making processes in the context of IoT technology adoption in DevOps-enabled environments.

1.4 Significance and Importance the Research

With increase in number of IoT devices, managing and scaling infrastructure becomes increasingly challenging. Problems concerning deployment and management complexity, scalability problems, security susceptibilities, large data volume management and analysis, as well as the topic of continuous improvement and innovation persist. Moreover, secure deployment of a significant number of IoT devices and hardware is a hard task. Update of software in a bulk on many IoT devices and tracking them at once is a very complicated task.

(Battina, 2017) Found that technology professionals also are concerned about security when working with technology. Rapid software changes that are done without the input of the security team will be prone to security issues. Many software developers and operations people feel that the typical DevOps roles such as automated monitoring can improve the security of systems.

Firstly, technical readiness is essential. The organizations should evaluate if their current IT infrastructure and development methodologies are able to support the increased requirements of the IoT systems including real-time data processing and management of thousands of IoT devices. This includes not only the functionality of hardware and software but also the skill of personnel in working with and blending these technologies.

Second, culture that exists in the organization is a big determinant of whether any new technology will be adopted or not. DevOps itself brings about a drastic cultural change for creating seamless collaboration between developers and operations teams. IoT addition to this equation necessitates additional cultural changes to accept new workflows, roles, and persistence in learning needed for the most effective application of IoT.

It is a well-known fact that DevOps has eliminated many problems of software development and delivery. But, in the case of IoT, which represent the combination of the software, hardware and firmware, very little has been written about it. Adoption of IoT in DevOps environments can enhance security by automating security measures, continuously monitoring for threats, promoting secure deployment practices enhance operational efficiency, innovation, and sustainable development, benefiting various industries, societies, and the world at large.

Therefore, the study seeks to determine the Factors influencing the adoption of IoT System in a DevOps enabled Environment.

1.5 Purpose of Research and Questions

The research study's aim to understand the differnent factor that affects the adoption of IoT System in a DevOps enabled environment. To fulfil the study's objective, the following research

1. What is the impact of the exiting support systems and resources within the organization on the adoption of the IoT systems in DevOps environment?

2. What do peers and leaders' opinions and behaviors influence on the IoT technology adoption decision in DevOps environment?

3. How do expected advantages and perceived demands influence organizational decisions to adopt IoT within DevOps frameworks?

4. What are the primary security issues related to the integration of IoT systems into DevOps environments, and how do organizational stakeholders recognize these issues?

5. What is the impact of perceived risks on making process in a context of IoT technologies adoption in DevOps-enabled environment?

Chapter 2 : LITERATURE REVIEW

2.1 IoT Landscape and Significance:

2.1.1 IoT Importance in Tech

(Kokila & Reddy, 2024) The Internet of Things is a combination of various technologies, including Machine-to-Machine (M2M) communication, Radio Frequency Identification (RFID), Wireless Sensor Networks (WSNs) and Supervisory Control and Data Acquisition in addition to big data analytics, cloud computing, and embedded systems, among others. M2M technology, which is commonly known as the Internet of Devices, captures events using network connections and sends these events to a central server, which interprets the events meaningfully to a particular application. The advancement of IoT devices globally collects tremendous numbers of data that require smart storage and management. Large volumes of data are processed, stored, and managed via big data technologies within diverse forms arriving at varying speeds. In addition, cloud computing offers capabilities over the internet and provides on-demand applications and workloads. The benefits provided by cloud computing include Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS), among others in addition to other platforms. Embedded systems control larger systems through the use of smaller hardware-based processing units that contain memory and networking. IoT connects objects to the internet as well as to each other, increasing the interaction of humans with technologies through innovative data sharing and exchange. Adverse interaction of technologies with humans will also develop a new phase. This interaction will make users more aware and responsible for their usage of digital products. For instance, “Product as a Service” models will be enhanced by the sensor’s technology.

(Rose et al., 2015)

The Internet of Things (IoT) covers a huge variety of products, services, industries, and socio-economic aspects. It centers on the widespread installation of the Internet lines and modern big

data collection into different items like smart goods, vehicles, industry parts, and sensors. These integrations into our lives area may help trigger the transformation of our way of life, workspaces, and hobby fields.

The forecast available show the possibility of IoT being a wonderful game-changer in the global landscape and economy. By 2025, experts make a forecast that the IoT devices around the world will achieve 100 billion connected devices, and the economic impact anywhere will possibly gain $11trillion. Huawei estimates that by the year 2025, the IoT connections will increase by 100 billion. Furthermore, the McKinsey Global Institute predicts that it could cause the budgets of the economies to fluctuate around $3.9 trillion to $11.1 trillion, globally, by the 2025.

Indeed, among high-impact applications of Internet of Things lies the possibility for usage in mostly developing economies. It is worth noting that the McKinsey Global Institute estimates that 38% of the IoT's economic impact by 2025 will be from less developed or East & which is about $5.7 trillion. This projection not only shows us significant growth and development prospects, but it also brings us closer to the dream of shared prosperity.

From the economic side of things, different variables are forecast to push forth these opportunities. Most of the resources continue to be consumed by leading countries, China particularly; developing countries by contrast are expected to consume a lot of these technologies in the future. Besides that, global expansion of the economy is mostly concentrated in the developing countries nowadays. However, industrial internet of things systems, again and again, will become a key technological component in manufacturing processes, construction sites, and transportation.

(Kumar et al., 2019)

Smart cities are a major field of application for the Internet of Things (IoT) and smart homes are one of its many aspects. Smart homes combine IoT-equipped appliances, heating/cooling systems, entertainment devices and security systems creating an integrated environment that aims at improving comfort, safety, and energy efficiency.

Smart cities have become a highly popular notion in the last few years, which stimulates intensive studies. Smart homes not only improve in-house comfort but also benefit homeowners by saving them money since they use less energy thereby reducing the electricity bills.

Apart from the smart houses, smart cities to some extents are also about smart vehicles. Modern cars are fitted with smart instruments and sensors that steer and monitor many components from headlights to engine activities. Smart car systems are now in a rapid advancement due to the intervention of the IoT technology, integrating wireless communication for car-to-car and car-to-driver interactions. Such systems are designed to facilitate predictive maintenance and to provide a more safe and comfortable driving.

2.1.2 Trends and Patterns

(Lampropoulos et al., 2019)

The IIoT is a particular area of the implementation of the IoT for the industries of today. It is also like a bigger machine that has many components. This is a fundamental element of the industries, and it is tightly linked up to the 4th Industrial Revolution, like Industry 4.0. IIoT is a bundle of services and technologies that use sensors, applications, software, and data storage for companies to be able to monitor as well as maintain their processes and equipment. The companies and the entire manufacturing system will face the slow work and the unreliable outcomes if these IIoT tools are not employed. In the I4.0 world, IoT including IIoT can be

synergized with cutting-edge technologies including cloud computing, big data, and CPSs to create a smart manufacturing. With the help of industry 4.0, the machine is capable of doing data collection and analysis since there is no need for human input in that task. It translates to the fact that the machine will be in charge of itself, perfect its performance, learn from its actions, and have the ability to adapt to different conditions which makes smart and efficient industries.

(Aman et al., 2020)

The comparison between the initial years and the most recent years clearly demonstrates a notable rise in the number of IoT reviews across all domains. The biggest increments, above 100%, were observed in the areas of Application, architecture, communication, technology, issues, and security. The percentage increases for the following areas over the previous decade are remarkable: application (483%), architecture (460%), challenges (111%), communication (760%), security (300%), and technology (343%). Communication was the most studied subject of research in 2017-2019, whereas application was the second. During this time, technology, security, and architecture were almost equally discussed. The IoT is characterized by four major attributes: heterogeneity, dynamics, scalability, and interoperation. Within IoT applications, three major classifications stand out: transport, healthcare, and smart surroundings. The healthcare industry is highly appealing for IoT applications, capturing the interest of the public, researchers, and companies. The advent of the Internet of Things (IoT) has yielded numerous benefits in terms of patient well-being, satisfaction, and the management and operation of hospitals. Various Internet of Things (IoT) technologies, For example, wearable devices, which connect with large amounts of data, cloud and fog computing, and make use of wireless body area networks (WBAN) and RFID, play a crucial role in healthcare. They offer adaptable solutions for mHealth applications and monitoring systems that facilitate functions like as ECG, blood pressure measurement, and oxygen saturation. The key areas of

interest in IoT environmental applications encompass waste management, climate and weather surveillance, and intelligent agriculture and farming. Waste management is a pressing global issue that distinguishes between the collecting of rubbish and the activities involved in treating waste. Climate monitoring also helps in the provision of weather forecasts that are important in saving life and property. Smart farming and agriculture result in higher efficiency at lower costs.

(Khan et al., 2021)

In contrast to major IoT applications, IoT is enjoying relevance in the process industries, agriculture, food and beverages, and supply chains among others. In process industry like petrochemicals, the continuous monitoring of key equipment parameters such as temperature, humidity, pressure, and flow is very important. By means of sensor-powered data gathering and analysis, predictive maintenance approaches can be applied. For example, keeping the tab on the temperature of cooling and heating fluids in heat exchangers allows detection of possible faults at an early stage. Likewise, condition monitoring solutions are applicable to centrifugal pumps, food storage vessels and winding machines among others to predict failure and reduce downtimes.

In agriculture, the IoT technologies make possible the intelligent resource usage. Data collected by sensors that monitor water levels, soil properties, precipitation, weather patterns, and crop production can be utilized to train machine learning models capable of forecasting crop health. This analytics-based strategy also allows for the optimized utilization of disinfection sprays and water. IoT-enabled drones can improve data acquisition and tasks such as water spraying, thereby, increasing agricultural productivity further.

As per **(Routh & Pal, 2018)** companies, which take the plunge into IoT implementation, without a solid business model form, are consumed by this vast lack of business understanding.

This model must take care of different e-commerce needs, it has to work for both vertical and horizontal markets and also have to satisfy different consumer preferences. In the opinion of the Harvard Business Review, companies began IoT initiatives with great expectations, but true obstacles limited the success. Each third qualifies as a misfire due to the Cisco research showing these statistics. Establishing them as the first choice in the insurance sector as the IoT is adopted will be made possible by Cognizant IT services by reworking roles, forming partnerships and preparing for future challenges, they emphasize the role of employees' acceptance and trust in the use of IoT tools, methods, and the maximization of the ecosystem's use. Interoperability is of course important, the report of McKinsey & Company predicts the creation of great survival ability for future IoT application, which the economic value ranges from $3. 9 trillion to $11. According to a recent World Bank report, the World's cities are expected to counter 1 trillion annual deficits by 2025. The Economic Times states that B2C use cases The primary catalysts of the IoT due to the large customer base, whereas B2B use cases are the main accelerators of the industry growth, with most of the commercial opportunities lying in the technical infrastructure, particularly in the area of connecting "things" and "data" generation.

2.1.3 IoT Architecture and Protocol

(Gupta & Quamara, 2020)

The implementation of IoT relies on the combination of several key technologies, including “Wireless Sensor Networks (WSN), Radio Frequency Identification (RFID), Machine-to-Machine (M2M) communication, and Low Power Personal Area Networks (PAN). Various frameworks have been suggested to comprehend their function on the Internet of Things (IoT), including those put forth by Global organizations and collaborative teams such as ITU, IEEE, Cisco, and ETSI. Nevertheless, these frameworks consider factors such as application needs,

network architecture, protocols, and business models. However, it is important to note that standardization is not yet fully achieved.

Since IoT is used across healthcare, transportation, and industrial management among others each with its industry standards and specifications, security emerges as an important aspect of creating dependable systems and applications Furthermore, the IoT design must be resilient in order to address various challenges that arise while implementing extensive networks consisting of diverse devices with limited capabilities, which connect instantaneously. The crucial aspects for promoting the creation of systems that deliver functionality reliably and efficiently are Quality of Service (QoS), modularity, reliability, semantic interoperability, privacy management, and support for new device kinds and services.

(Mrabet et al., 2020)

IoT could be seen as a compound of several basic technologies varying the suggested architecture at different times. On the other hand, depicted architectures mostly include three-layer, middleware-based, service-oriented, four-layers, and five layers models.

The three-layer architecture consists of the perceptual layer, the network, and the application layer, ensuring that none of these layers are overlooked. The architecture comprises the display layer, business layer, service layer, and data layer, which collectively serve as an intermediary for administration, data storage, and composition. Next in line the five-tier architecture has a part of the object abstraction and business layers.

For the IoT while the IoT data management capabilities, machine learning and encryption among other are given more prominence, a five-layer architecture approach is highly useful. This design comprises two levels: the perceptual link and the protocol network layer, along with the transport application layer and the data/cloud services. The physical layer encompasses a range of sensors and IoT devices, including Wireless Sensor Networks (WSN),

QR Codes, and RFID. The network/protocol layer contains protocols that connect to the wireless and wired network hardware like Wi-Fi, Bluetooth, and 5G. The cause is the TCP/IP and TLS/SSL protocols which belong to the transport layer. IoT is solely exclusively completed by the application layer by the protocols like AMQP, CoAP and MQTT. Finally, data/cloud services layer combines to build an interconnecting framework that are cloud based IoT.

(Alhaidari et al., 2023)

The more data we accumulate from connected devices, the more we will need to use additional storage to keep it all. This data can give us useful information. However, dealing with this data needs extra power of computation, which is what Cloud Computing provides with its vast storage and processing powers. Admittedly, we haven't started utilizing this in IoT completely as yet, but IoT devices can tap into the enormous storage and computing power that cloud servers have to offer, which is referred to as "Cloud of Things" (CoT). The CoT architecture allows us to integrate various components as well as layers which means Cloud computing enables the processing of, storing of, and manipulation of the IoT data. It's made up of four layers: We are happy to provide the $700,000 credit to our customer in Spain. Sensing Layer: This tier collects information from many sensors, organized by the type of sensor that they are. 2. Communication Layer: This layer plays the role of the main network to the IoT, connecting all the different IoT bits, such as sensors, devices, and humans. 3. Control Layer: In this place we keep and deal with data of any IoT thing and provide services to such gadgets – watch over and service them. 4. Actuation Layer: This layer accepts the information we get from the control layer and transforms into something that can be used by either people or machines to do things automatically. To put it in layman's terms, the CoT system helps to enhance the utilization of cloud computing for enhancing the functioning of IoT by controlling the activities of data processing and communication from different parts.

(Khan et al., 2012)

These days, with the crazy technology, the smart devices got many things improved like the processing power and storage capacity, and the size is reduced too. These devices, which are now equipped with various sensors and actuators, have the Internet connection and communication features that enable them to connect with other devices and hence the Internet of things opens up a world of possibilities for the future. Besides, physical objects are now being equipped with RFID tags or electronic barcodes which enable bi-directional communication with smartphones or embedded RFID scanners. The Internet of Things (IoT) was born in 2005, a concept that intends to network the real-world objects in a sensory and intelligent way. A prime example of a basic IoT system is the interconnected things that communicate to do various applications or services. The IoT workflow is made up of objects detecting, identifying, and sending the data they are going to process, carrying out the necessary actions without human intervention, and giving the administrators with services and the information they need. This basic process emphasizes the main features of IoT, where the devices interconnect to collect data, to make decisions and to provide service to different domains.

(Sethi & Sarangi, 2017)

The IoT Network Protocol Stack comprises various layers designed to facilitate communication among IoT devices using IP, recognized for its flexibility and reliability. The IPSO Alliance, in collaboration with the Internet Engineering Task Force (IETF), has created another protocols and standards for various layers of the IP stack. They have also built an additional adaption layer to improve communication between smart objects. The IEEE 802.15.4 protocol facilitates communication between low-power embedded devices at the Physical and MAC Layer. It emphasizes long battery life and cost-effective, short-range communication. Although the device has a low power consumption of one milliwatt, its communication range is limited.

However, by utilizing cooperative device functioning, it becomes possible to achieve multi-hop routing across extended distances. The protocol supports small frame sizes, low bandwidth, and a maximum communication rate of 250 kbps, with built-in redundancy for robust communication and error detection. Additionally, the adaptation layer, such as 6LoWPAN, facilitates IPv6 communication over low-power wireless networks, bridging the gap between bulky IP protocols and resource-constrained environments like IEEE 802.15.4. With IPv6's scalability and extensive addressing space, 6LoWPAN enables communication with other IP-based devices on the Internet, albeit requiring gateways for connectivity due to IPv6 header size constraints. This adaptation layer optimizes packet transmission by squeezing and fragmenting IPv6 headers to fit within the IEEE 802.15.4 standard's MTU limit, ensuring efficient communication in IoT environments.

(Uy & Nam, 2019)

The standard data transfer procedures that are commonly used in IoT systems are MQTT, CoAP and AMQP. MQTT and CoAP were all studied and compared before, where MQTT is usually the winner in many cases. Nevertheless, AMQP is a protocol that is frequently used for data transmission between IoT gateways and servers. The author is of the opinion that he has reviewed the entire AMQP and MQTT in detail, outlining their pros and cons and the usage of both depend on the system requirements of the 3-layer Internet radio. AMQP is the best in the field of effectively delivering data from devices to software applications through the use of queues. In contrast to MQTT, where the messages are directly sent to the consumers by the broker, AMQP messages are first, transferred to an exchange, then the exchange decides which consumer it will send the message to, and the message is then stored in a queue. While MQTT is praised for its simplicity, efficiency, and the fact that it is suitable for M2M, WSN, or IoT applications, MQTT is, on the contrary, the one that is not liked for its simplicity. It needs fewer network resources; hence, it is suitable for the case of low loss rates environment. The

MQTT's tight connection with the efficient transmission and the minimal use of the resources makes it the most suitable for the situations when the data packets are not continuously sent, and the loss rates are moderate. On the other hand, AMQP has more advanced features like the flexible routing, the durability, and the high availability queues which makes it more reliable and secure. The glue is AMQP which is a longer protocol, but it is very good in situations where continuous data transmission is needed or when the data loss rates are either very low or very high, thus, showing its strengths in data integrity and delivery.

2.1.4 Overview of IoT Hardware and Software

(Bîrlog, Borcan, & Covrig, 2020)

IoT systems are characterized by hardware components that are the control units, dashboards, routers, bridges, servers, and sensors. These gadgets are charged with various activities which are among the threat detection, system activation, security checks, and executing the specific actions. The foundational elements of IoT hardware include: The asset to be monitored or controlled, the data acquisition module, the data processing module, and the communication module are the components to be interfaced for a given system. The asset to be monitored or controlled may be either a standalone device or a smart device that is integrated into the smart system. The module of data acquisition is in charge of obtaining the signals from the outside world and translating them into digital signals. The primary components of the sensor system are sensors for the collection of real-world data which include light, temperature, vibration, or pressure, as well as the necessary circuitry for signal conversion. Furthermore, the data processing module is the principal component in an IoT device. It receives incoming data, performs needed operations, and retains the processed information. Some of the IoT gadgets have processing abilities inside them, which allows them to process the data locally before sending it to the cloud, while the others depend on the gateways or cloud applications for the data storage and processing. Besides, the communication module acts as a communication

bridge which links the components with each other and the cloud platforms. It is the medium which transmits the data both to the centralized systems upstream and to the other IoT devices or applications downstream. Collected, these hardware parts are the foundation of IoT systems that allow for the efficient data acquisition, processing, and communication for different purposes**.**

(Maier, Sharp, & Vagapov, 2017)

The need of powerful, cheap, and energy-saving solutions for IoT devices is very important in order to increase and widen the application of IoT. A necessary condition for these tools is their small size, because the smaller and lighter the device is, the more applications it can be used for. Generally, an IoT device includes a microcontroller (μC) and a wireless communication module, usually Wi-Fi. There is a great variety of modules and microcontrollers on the market but some of them are either too expensive or too big and heavy. Also, the problem is that only a few modules are open-source and there is no limitation on their operational purposes. ESP32 is the successor to the ESP8266 μC and tries to solve these problems. The ESP32 has the built-in Wi-Fi and Bluetooth the ESP32 QFN48 is much smaller than other microcontrollers like Xbee or RTLDuino, and its footprint is only 5mm x 5mm. The ESP32's circuit, for example, the ESP-WROOM-32 module, ensures the easy integration into custom PCBs which in turn enables the creation of space-saving devices. Dual-core Harvard Architecture Xtensa LX6 CPUs, the ESP32 provides a classy performance. The programs, real-time operating system (Free RTOS) and development framework (ESP-IDF) that are embedded in it make it even more suitable for IoT applications. Although MSYS2 is designed mainly for Linux, there are tools such as MSYS2 that allow ESP32 development in Windows. The ESP32 is a good choice for various IoT projects, due to its flexible form-factors and reliable performance, thus, it serves for hobbyists, students, industrial manufacturers, and small-sized solutions.

(Lekić & Gardašević, 2018)

Node-RED is a flow-based development tool that is open-source and highly regarded by IBM Emerging Technology. It excels at connecting IoT hardware devices, APIs, and online services, and its primary objective is to ensure a smooth and effortless integration process. This JavaScript tool, developed in a Node environment, serves as a prime example of effective utilization. The JavaScript platform is an online platform that features a visual browser-based flow editor. This editor utilizes nodes represented by icons, which are designed to be easily comprehensible. It has both the drag-and-drop node wiring and JavaScript code importing at its function. Nodes in Node-RED are the elements that have several functions such as debug out nodes for flow monitoring and GPIO pin interaction on Raspberry Pi devices through nodes specifically designed for the purpose. Flows that are made using Node-RED are stored in JSON format, thus the developers can connect the input, output and the processing nodes for different purposes like the data processing, device control or the alert notification. The fundamental idea of the tool is to make the connections between the web services, nodes, and IoT devices, and to perform the tasks like sending the sensor data to the email or social media platforms, and to analyze the data. Node-RED comprises three fundamental components: The Node Panel, Flow Panel, Info panel and Debug Panel are the examples of the section that is composed of the flexibility and robustness panel for the prototype development, which is especially useful in the event-driven systems like the IoT applications. This tool is demonstrated by the following practical example which involves a Raspberry Pi 3 model B and a DHT11 sensor combined with IBM Bluemix Cloud, and thus it is shown that it can create real-time IoT applications that are easy to use. Using Node-RED and the WebSocket Protocol, developers can create flows that will make data transmission and command processing easier and also help to collect and store big data from IoT systems. Moreover, Node-RED has the abilities of data retrieval,

processing and analysis which provides the base for a wide range of application scenarios and IoT concepts, and this will be the field for the future research and experimentation.

2.1.5 Current Challenge in IoT

(Khanna & Kaur, 2020)

The challenges of IoT are therefore diverse and pervasive to both research and industry aspects. Mapping, meaningful communication in heterogeneous environments, and robust communication technologies are among those challenges. The network technology needs to perform the task of embedding physical objects into the Internet effectively, while the network discovery is confronted with the dynamics of changes. Semantic interoperability and service discovery are among many problems that data management raises. The standardization plays a major role to support various applications and typical needs.

Major challenges are as follows: maintaining the total information service, uninterrupted network connectivity, operational continuity, security issues, ongoing services effectiveness, and standardization & proper identification. These hurdles cut across many fields and manifest a dire requirement of continuous improvements and renewals.

(Kumar et al., 2023)

The IoT integration in healthcare holds great promises and one of the areas where advances are expected is in data collection and analysis. Still healthcare providers are faced with the challenges of managing large amounts of data that are being produced by the IoT devices. The author stresses the necessity of operational information processing systems related to data collection, storage, analysis and dissemination. However, to deal with these imperatives, the mind sponge theory recommends the use of advanced analytics and machine learning algorithms to prioritize the right information and improve care outcomes for the patient.

(Mohd Aman et al., 2021)

The growth of IoT systems has been impressive during the era of the industrial revolution. Nevertheless, battery-based electricity provisory for IoT devices come with challenges on energy resource management. These devices have specific energy needs, fueling frequent battery changes and high energy demand. The infusion of smart IoT systems in electric cars can result in enhancement of traffic flow and road safety. However, the control of energy management of the integrated electrical systems is a critical problem, as classical controllers have some operation and reliability constraints.

Smart energy management problems can be classified into six issues that include integration, consumption, conversion, communication, multifunction, and stability. Integration comes with the challenges of diverse hardware, software, and middleware entities of each device, which is required to be converted at various levels, prices, and sizes. The use of renewable energy systems is more variable with instabilities that are caused by environmental variations while device functionality and objectives influence energy consumption.

Energy storage and management are major proponents that underpin energy-efficient operations as the demand for smart applications leads to the connectedness of IoT objects. This is more crucial as many IoT power systems depend on batteries with short life, stressing on the necessity of successful energy management strategies.

(Tawalbeh et al., 2020)

Although IoT brings users a number of advantages, it raises several issues as well, with cybersecurity and privacy as its main problems. One of the challenges that IoT devices bring is the high level of security issues that they attract due to their specialty, massive production, and the similarity of deployed devices in terms of security issues. While implementing IoT, the security challenges need to be addressed as a part of building trust with the consumer and an

effective protection against cyber threat and data breach. The weak protection of IoT devices and services makes them the frequent victims of cyber-attacks emphasizing the necessity to improve security measures. The other obstacle for the full integration of Internet of Things is also privacy. The users are more and more suspicious of the possible privacy violation and the exposure of the date. Adherence to user privacy rights is crucial in building trust in IoT technologies and services. There are several factors that can undermine the security of IoT devices such as infrequent updates, hard-coded passwords, automation vulnerabilities, remote access protocols, third-party application risks, improper device authentication, and poor device monitoring. However, tackling these challenges demands proactive approaches to improve security procedures that preserve the integrity, and confidentiality of IoT systems and data.

(Dofe, Frey, & Yu, 2016)

Globalization of the chip supply chain has introduced fears concerning chip security, assuming that chips ought to always operate properly even when purposely attacked. Because a growing number of different firms are involved in the design and production of chips, possible motivations, or situations for any of them to tamper with the respective chip are plentiful. Several examples derived from the real world, including “kill switch” and chip backdoors, as well as hardware Trojans in commercial goods, disclose the necessity for the consideration of hardware security in future IoT applications. HTs are changes to the initial chip design that cause the chip to operate incorrectly. Neutralization strategies include both destructive techniques, such as budding-in, by components such as chemical mechanical polishing, as well as non-destructive techniques such as the comprehensive investigation of the chip power and delay. A countermeasure, PUF circuits, is proposed, which offsets the threat of malicious insertion in the IoT node. Moreover, existing Electronic Design Automation research, addressing essential issues including trust policy and security averaging, conclude that EDA operations should continue to be executed in conjunction with those addressing practical

objectives such as the IoT applications. Further, due to the fact that attackers can decrypt secret keys by analyzing side-channel signals, countermeasures against side-channel analysis must be considered. The SCA attack category encompasses Side-channel analysis attacks, Simple Power Analysis, Correlation Power Analysis and Differential Power Analysis. Among these, Correlation Power Analysis is particularly severe because it requires fewer traces to determine the key compared to the other two. Additionally, an attacker might employ HTs as well as a CPA assault to access data or disrupt service provision in IoT. Thus, while researching whatever methodology, one ought to seek integrated solutions whereby each target, if possible, should be covered.

2.2 IoT and Security

2.2.1 Overview of IoT Security Challenges

The Internet of Things (IoT) raises a number of security risks and can be broadly categorized into three areas, according to **Iqbal (2020)** IoT data, communication, and end applications associated security. The IoT has made it possible for physical systems and items to communicate at a previously unheard-of degree, creating new security risks. A major worry is the enormous volume of data that end nodes in the IoT ecosystem create. This information may be sensitive or private, making it useful to potential attackers and business rivals. Despite the importance of maintaining the confidentiality of this data, IoT nodes' resource constraints prevent the use of standard encryption algorithms. Therefore, there is an urgent need for lightweight cryptographic ciphers that can offer the best level of confidentiality in nodes with limited resources. For end nodes, ensuring authenticity presents another difficult problem. End nodes' authenticity can be compromised through physical assaults like hijacking, and node copying, replacement, making them vulnerable to attack.

Yet, the security question of the communication in the IoT remains also one of the most significant. For the purpose of guaranteeing the security of the communication network,

authentication and access control of devices and users in the IoT environment are the main demands. End applications, the last stop for data collected from IoT devices, pose a serious security challenge. End nodes supply IoT gateways with huge amount of data, which is then spread across different IoT systems and networks. The flow of the resultant data brings concerns about privacy, legal and social aspects, forensics. Also, the private information of a user can be used to as an identifier, like the heartbeats, fingerprints, and other environmental characteristics sensed by the end nodes for the user tracking and preferences. Besides, user monitoring and preferences can be identified using private or personal information of the user, for example, heartbeats, fingerprints, and different environmental characteristics sensed by end nodes. This also raises major privacy issues and highlights the necessity of designing privacy protection into IoT systems. To sum up, security problem in IoT is multifaceted and challenging that need a comprehensive approach to overcome many security issues in each phase of the IoT data generation, transfer, and usage.

The expansion of smart IoT system and the Internet of Things (IoT) has changed the way in which we interact with our environment, however, it has also created new security problems, argues.

Elena and Anna (2022)

Due to the fact that a large number of smart devices have a finite computational power and resources, it is difficult to guarantee security of IoT systems. As such, they may not be as well-endowed as non-IoT devices to perform intensive security functions that require a lot of resources, leaving them more susceptible to intrusion.

One of the primary challenges regarding the security of IoT is the absence of ample processing power for efficient embedded security. Most smart devices have low resources and can’t

support heavy security software. This complicates the development of strong security features in the firmware of the device making them susceptible to different types of attacks.

Faulty access control is another significant problem in IoT security. A lot of IoT systems are created to be user friendly and little attention is paid to the access control. This also makes it easy for the hackers to get hold of private information or remotely take control of a device. Inadequate access control also makes it difficult to recognize and rectify security problems.

A challenge in the field of IoT security is also a lack in funds for testing and enhancing firmware's security. Most IoT devices are made to be cheap, with poor resources for firmware testing and improvement. This refusal to invest in security testing leaves the termite to go undetected, making it easier for attackers to take advantage of them.

The absence of regular patches and updates is yet another problem in the IoT security. Paradoxically, the availability of software upgrades may be periodically limited by the technological and budgeting aspects of IoT devices. This may also happen if users forget or refuse to update their devices, in turn leading to known security issues. What is more, with time, software updates may not be available for the older devices, which may in turn make them vulnerable to attacks.

The physical assaults are a threat in the IoT security as well. This type of attack is hard to identify and prevent what makes it the major problem of IoT security.

Finally, to hack the communication of a target IoT system, infect it with the malware and theft of the sensitive information, the criminals are going to exploit the vulnerabilities they have found in the system. For example, to hack the Ring smart cameras, they use weak, recycled, and default passwords. They have also employed the microphone and speaker of the camera, which can be considered as an illustration of the dangers of the security vulnerabilities of IoT.

(Zhi-Kai,2014)

Emergence of the Internet of Things (IoT) has enabled the interconnection of billions of devices, transforming the way we live and work. However, this technology has also brought with it new security issues, with IoT devices being prone to intrusions by hackers and malware. The advent of malware that targets IoT devices is one of the biggest challenges to the security of these devices. IoT devices' constrained resources make it difficult to protect against malware using conventional security measures. The computational capacity of IoT devices is very limited., despite antivirus software being one of the best instruments for identifying known viruses in the real-time paradigm. Antivirus software's real-time scanning feature can cause IoT devices to incur exorbitant costs. Additionally, malware authors design their malware into distinct downloader and main body components by taking into account the IoT's computational power issue. Additionally, the diversity of hardware designs among different Internet of Things devices makes it difficult to provide a general abstraction of IoT malware. In the absence of a comprehensive conceptualization of IoT malware, existing remedies may be improvised or even irrelevant. Therefore, there is an urgent need for security systems that are tailored for IoT and that can successfully counter the threat of malware that targets IoT. Such security methods must be capable of identifying malware in real-time without imposing an exorbitant overhead while taking into account the constrained resources of IoT devices.

2.2.2 Existing Research on IoT Security and Penetration Testing

According to **Shakdher, Agrawal, and Yang (2019)** IoT applications face different security problems that could put security at risk. Code injection, flawed authentication, unencrypted sensitive data, flawed access control, security misconfiguration, and reflected XSS are six vector attacks that the author of the study names. Such vulnerabilities are used to get our unauthorized access to our confidential data that cannot be of good consequences.

A comprehensive penetration test was carried out by the author using popular techniques such as DNS lookup, MITM, DNS cache poisoning, and denial of service (DoS) attacks to examine the weaknesses in IoT apps. Many web and mobile IoT apps are usually applications used in industry, such as connected cars, smart homes, security systems, and the healthcare industry. Many of these applications have over 1 million of a user base and all with a high effect in terms of the harm they can cause if compromised.

The results of the study indicate that many popular IoT applications, even though they try to secure their connections via HTTPS, are vulnerable to attacks. Among the principal reasons of this vulnerability is wrong security parameters adjustment. For example, in the first attack vector, the writer succeeded to move a secured SSL connection to the insecure one through the use of IP forwarding, sniffer, ARP spoofing, SSLStrip+, and DNS2Proxy, thus, the sensitive information such as usernames and passwords were exposed.

The research states that more attention should be paid to security issues of IoT application and proper security configurations are necessary in order to avoid such vulnerabilities. With the growth in use of IoT devices in day-to-day life, it is very important to make sure they are secured and not easily exploitable by attackers.

In one research paper, **(Garg & Dave, 2019)** introduced a secure IoT architecture that allows ensuring end-to-end security from IoT applications to IoT devices. Integrity of each component allows to evaluate the system security. IoT devices are isolated as they are tied to a gateway, which acts as a mediator to the internet interface and does not have any connection with the outside world. Consequently, a few firewalls will help in securing equipment against hackers.

The IoT gateway act as a link between middleware and IoT devices, it allows data to be transferred securely through REST API calls. Communication between the IoT gateway and the middleware is secured using standard cryptographic methods since none of the parties is

resource constrained. Authentications and authorizations are taken care of by REST APIs, thereby, making the process simpler and meet the industries' standards.

The author also proposed a middleware architecture that provided sensing data submitters a complete security solution. This approach allows for the whole data to be encrypted in order to protect it in transit. The proposed middleware and gateway solution use REST API for data exchange and communication and consider all limitations of IoT systems. The middleware helps in the development of IoT by making REST APIs available and providing consumers the way to register their IoT devices and to safely access the data from them.

(Johari et al., 2020)

Emphasizes that multiple penetration testing methodologies need to be considered for protection and attack scenarios. It is observed by the author that although users are often concerned with protecting their applications from attackers, they rarely consider how to address important questions such which methods were used for the attack, what the goals of the attackers were, and how the attack was organized.

Among other factors, the complexity that is required to find the vulnerability, execute the attack, and stop it, can provide a measure of the success of the attack, the author argues. Feasible methods of attacks are web implants, viruses, SQL injection, password guessing, and brute force attacks.

Once an attack is revealed, the author recommends conducting penetration tests with access to practical measurement tools that help in discovering and evaluating network vulnerabilities. In the end, penetration testing ensures the network is secure because by discovering and fixing vulnerabilities before they can be utilized by attackers.

(Chandan & Khairnar, 2018)

In the IoT Penetration testing world, firmware cracking is a critical process after the testing of an application, mobile device or IoT device. This is because the firmware is often poorly coded and built, which can lead to critical credentials and port information being stored in cleartext. This important data is readily available since the firmware is sometimes posted online, or is packaged with IoT devices, and attackers may use it to their benefit. So, while testing IoT devices,

The above testing methods' failure makes hardware cracking the final option. Despite the fact that it is a tedious process, as it involves the removal of the hard disk and other memory chips which contains the microprocessors and microcontrollers which stores personal data that can be read and accessed by a hacker. There are cases when totally breaking a hardware is irreversible and that component will be not mendable.

The tools used in firmware cracking in IoT penetration testing are diverse and each tool is used for application in particular test cases. One example is Firmadyne, a customized automated system designed for firmware dynamic analysis and emulation of embedded Linux firmware. Another example is Binwalk, a firmware analysis tool that aids in analysis, reverse engineering, and extraction of firmware files. The Firmware Analysis Toolkit (FAT) includes several emulation tools and application such as Firmadyne, Binwalk, Firmware Mod Kit, MITM Proxy, and Firmwalker. QEMU is a firmware simulator..

To sum up, IoT pentests require hardware and firmware cracking and such tools are manifold. Nonetheless, hardware cracking should be conscientiously carried out as it can render the device unusable.

2.2.3 Case Studies of IoT Security Breaches 2.3

(Alladi et al., 2020)

This case study illustrates the exposure of Internet of Things (IoT) devices and the possible impact of an attack. Here, attackers managed to hack into a Tesla Model S by using weaknesses in the Tesla Service Wi-Fi SSID, QtWebkit/2.2x browser engine and the Linux security module – AppArmor. The attackers managed to gain control of the car remotely in idle and driving modes by writing their modified software onto the Gateway ECU.

Another vulnerability that Chrysler has shown is the 2015 recall of 1.4 million autos over alleged security issues. To avoid known security vulnerabilities IoT device manufacturers should regularly update and patch their devices in a secure manner, either via secure over the air wireless updates or an alternative method.

The Tesla Model S vulnerabilities were mainly caused by the absence of means to prevent the browser memory leaks, the missing checks for changing privileged user in the car, and the unsecure storage of secure tokens for firmware integrity checks. In order to prevent such attacks, AppArmor could be further strengthened by disallowing kernel address leaks, and enforcing strict access controls that forbid privileged directories to be accessed in the browser Furthermore, the Linux kernel distribution that Tesla uses can be altered to include access controls for modifying the rights of a privileged user and to patch known CVEs.

Session keys can be generated with True Random Number Generators (TRNGs) which disable the unsafe storage of the secure tokens and prevent unauthorized firmware access any IoT device.

This case study illustrates the significance of IoT device security and manufacturers should install measures that will thwart un-authorized intrusion. The numerous and diverse cyber interfaces located in IoT devices enlarge attack surfaces and security exposures. Hence,

security should be the main focus of IoT device manufacturers, who need to apply the measures in order to secure their devices against the vulnerabilities and attacks.

(Vanwell, 2021)

Ringing, which is a company owned by Amazon, has been in the headlines lately due to some problems associated with user data security and IoT security breaches. The first situation came up when the organization accidentally exposed user data to Facebook and Google via third-party trackers integrated into their Android application. In the second incident, the attackers exploited doorbell and home monitoring systems of a number of households associated with each other. By gaining access to the live feeds of the customers' houses' security cameras, the audio feed was hacked and turned against the customers using the devices' built-in speakers and microphones.

Weak, repetitive, and generic credentials were used during the IoT security attack. As most of the buyers of the new "smart" devices neglected to alter the default admin settings, the systems were open to hackers. Moreover, many people employ one login for all their subscriptions and accounts, making it easier to use credentials stolen from one provider in order to log into another.

These occurrences are an excellent illustration of why you should practice simple rules of cyber security like creating unique login details and also changing admin password when receiving new hardware. Since then, every Ring owner is instructed to use two-factor authentication, strong passwords that are changed frequently and added Shared Users instead of sharing their login.

The examples of the Ring security violations point to the importance to secure IoT devices that are likely to enter and to watch private places. The greater the quantity of devices that have access to the internet, the more operations users have to conduct to secure their personal data

and devices from cyber threats. Some of the simple actions that can prevent from security breaches and save personal privacy include changing of default passwords, creation of strong and unique logins and use of two-factor authentication.

In the end, Ring security incidents should serve as a lesson for all IoT device users to follow simple cyber security rules that will help to keep your personal data and devices safe from cyber threats. New password, strong unique login details have to be made and two factor authentication is very critical to avoid security breach, at work or in private life. By the above measures, users also can contribute to the fact that their IoT devices are kept in a safe and reliable state.

(Critchley & Latonick, 2020)

JSOF disclosed the Ripple20 vulnerabilities, which consist of 19 CVEs that impact the Treck TCP/IP stack.The remote code execution vulnerabilities are many and found across and provide the attacker a complete remote control over the target device. Finite State focused on the two principal remote code execution vulnerabilities discussed in the disclosure, CVE-2020-11896 and CVE-2020-11901. The team rapidly heard all firmware revisions for many devices containing the Ripple20 CVEs via focusing emulation approach to not to disrupt the network. Based on their investigation, the ratings of the two devices which were publicly demonstrated exploitations by JSOF were consistent with the scores indicated for CVE-2020-11896 and CVE-2020-11901. These CVEs were related to the different devices expected result changed, however. However, according to their study, the majority of the devices that use the Treck stack do not suffer from the newly publicized remote code execution vulnerabilities since the set up of the devices. According to this, the widespread impact of the threat of Ripple20 is doubtful. The reporting and analysis of vulnerabilities mechanism may also need some changes due to their discovery, given that the report asserts the necessity of verification of vulnerability throughout various versions of affected devices. Especially in terms of embedded devices,

which come as source code, where OEMs have an opportunity of modifying and choosing some parts of the code that provide stack functionality, the security community should develop scalable approaches to verify and solve reported vulnerabilities. Security teams have rushed to determine if their devices are vulnerable due to the fact that it is believed that these vulnerabilities involve a huge spectrum of devices used by all businesses.

(Sherman, 2022)

In November, the Dutch multinational company Philips reported some vulnerabilities in its TASY Electronic Medical Record (EMR) HTML5 system that may result in the disclosure, extraction, or retrieval of the TASY database patient's private data. Among them, there is a detection of three vulnerabilities in an MRI software solution later in the month. The worst part was when it was revealed that its platform and solutions for IoT medical device interfaces had weaknesses, which allowed the attackers not only to access the patient data but also to initiate denial-of-service attacks in case of success. Moreover, alerts were issued by the Cybersecurity & Infrastructure Security Agency (CISA).

There are serious flaws in the PIC iX, Efficia CM Series, and IntelliBridge EC40 and EC80 systems which seriously threaten patient data privacy and security that are protected under HIPAA. Since Q3 2021, Philips address the insufficient input validation in PIC iX C.03.06, and it plans to address the usage of a hard coded cryptographic key and the weakness in the unsecure cryptographic algorithms before 2022. This vulnerability is specifically worrying in a medical device, which deals with patient information because hardcoded passwords are easy to break.

Hospitals now have to consider their own attack surface that did not exist before because of the surge of IoT threat in the healthcare industry. The security plan of healthcare institutions should cover the protection of patient data. It is critical that medical devices are also secured and that

measures to protect against cybersecurity threats are in place. This calls for working together of healthcare practitioners at medical device cybersecurity specialists for identifying the points of weaknesses and introducing the best practices for securing medical devices.

To sum up, the latest security weaknesses in Philips medical gadgets are a serious threat to patient data privacy and security. The vulnerabilities should be fixed immediately in order to avoid data compromises and other cyber-attacks. Healthcare providers should work together with the cybersecurity experts to detect potential weaknesses and adopt standards for securing medical devices to guarantee patient data privacy and security.

2.3 Penetration Testing for IoT Systems

2.3.1 Definition and Purpose of Penetration Testing

The process of system or network evaluation for weaknesses using a number of offensive techniques is called pen testing or ethical hacking, sometimes referred to as penetration testing, as take pointed out by **(Yadav et al., 2019)** The goal of this test is to secure the sensitive data from other people such as hackers who could take advantage of the unauthorized access to the system. Penetration testing is a process where an authorized attack is simulated through a system or network to find vulnerabilities for an access to confidential data.

The aims of a penetration test are to determine whether the current defensive measures of the system succeed in preventing security breaches. Upon discovery of the vulnerabilities, they are used to execute an assault on the system, to steal the private information. The penetration test results are recorded in a report, which contains the identified weaknesses, their associated risks, and remedial recommendations.

Any company looking to protect its critical data from unauthorized access must carry out penetration testing. Through the detection of vulnerabilities and weaknesses in their systems or networks, organizations can implement required measures to lower the threat of cyber-

attacks. Penetration testing also helps firms to remain current on the latest security threats and to secure that their security controls are strong enough to repel any security breaches.

(Bacudio et al., 2011)

The modern organizations suffer serious financial and image risks related to security breaches. It is enough to see that recovery efforts alone will be $167,713 per occurrence that the expenditure that a module of the type causes is significant including the costs of notification, remedy actions, decrease of productivity and loss of revenue. But this cost can be minimized using a penetration testing, which helps to determine, mitigate and resolve risks before security breach incidents happen.

Penetration testing is crucial in view of the fact that non-compliance with industry-mandated regulatory requirements lead to very huge fines, imprisonment, or even bankruptcy. Penetration testing in such a case provides undeniable evidence which helps organizations in fulfilling the auditing or compliance requirements of the laws.

Compromised client data can result in a decline in customer trust and brand destruction of an organization. This could threaten the whole organization. On the other hand, penetration testing provides security awareness all over the organization, thus avoiding security incidents that could damage the company's reputation, corporate identity, and the customer loyalty.

In summary, penetration testing is a preventive service, which can avoid security breaches, minimize financial and reputational risks, and keep compliance with industry-based standards. Organizations would save money, maintain their reputation and loyal customer base if they identify and resolve risks before they turn into actual security breaches.

(Johari et al., 2020)

However, the Internet of Things (IoT) has also enabled great technology achievements but introduces significant security worries. Entry tests have been the main focus of the researchers when developing secure and vulnerability free software in order to address this issue. Software development life cycle (SDLC) model that highlights certain characteristics of penetration testing helps to understand how the product is created and is an important in understanding the concept of penetration testing.

From collection of requirements and identification of use cases which need to be addressed during the writing of the program, the SDLC has different parts. After this phase comes the detection of abuse situations, which are inappropriate or boundary cases that need to be considered during software testing or development so that negative feedback from end users can be avoided. The next stage is to define the security requirements, which are the operational security dimensions of the software.

After finalization of requirements and use cases, the good software design should be developed to define the sequence of processes for risk management on a project involves planning, risk analysis, risk identification, and risk control.. Test cases should be provided to make sure that no use case goes untested. Once the design, use cases, and test plans are ready, development may begin. After coding, the testing process is performed, and the results are registered as “PASS” or “FAIL” for each test case. End-user input is expected and appreciated after the software is launched.

Penetration testing is said to be the last phase once application development is finished, and a product is ready to be accepted. It is an important part of guaranteeing security of the software and absence of vulnerabilities. Using the SDLC model alongside penetration testing,

developers can craft dependable and safe applications that fulfill user needs and guarantee compliance with industry-wrought regulations.

In summary, penetration Testing is crucial for the development of secure software in the age of the IoT. The SDLC model serves as a guideline for the developers to ensure that the software is being developed keeping security in mind. Penetration testing should be included throughout the software development life cycle in order to detect and rectify security breaches before they are exploited.

2.3.2 Penetration Testing Methodologies and Tools for IoT System

According to **(Denis et al., 2016)** the four common forms of penetration testing include external, internal, double-blind, and blind testing. Acting on a company's outward facing servers or devices enables you to perform an external test to find out of and how much access an outsider attacker can get. An internal test recreates an attack, from a normal user with regular access rights, from the inside of the firewall. Blind testing emulates the actions and methods of a real attacker by withholding some information from person or team performing the test. But by allowing a few chosen people within the company that the test is being carried out, double blind testing extends blind testing a bit further. Overall, such penetration testing can help companies to locate gaps and weak points in the infrastructure of their organization, enabling them to take the necessary steps towards enhancing their security posture.

(Chowdhary et al., 2020)

Penetration testing current practice is mostly done by means of manual methods which involves scoping and reconnaissance, vulnerability analysis, and exploitation as well as reporting.

The first step is scoping and reconnaissance, where the pen tester scopes out a security assessment and uses reconnaissance tools like Nmap to scan the network.

The next technique is vulnerability analysis, where the pen tester exploits already known vulnerabilities in a target environment using tools such as Nessus and OpenVAS, or explores unknown vulnerabilities using tools like Burp-suite.

The third stage is exploitation and reporting, and the process is carried out through the use of custom scripts or tools such as Metasploit to exploit the vulnerabilities. The pen tester also collects proof and drafts a comprehensive report of their discoveries.

Nonetheless, there is a requirement for faster and automated methods to penetration testing to minimize the duration and effort in the manual testing. Some of these steps can be automated to deliver more precise and overall results and free pen testers to concentrate on other vital parts of the network.

(Jevtic, 2019)

All kinds of vulnerabilities such as network, application, and system, can be detected through penetration testing. Below are the top penetration testing software and tools for professionals:

Kali Linux is a Linux Based Distribution that primarily was developed for Penetration Testing. It is generally agreed as one of the most powerful password cracking and network infiltration tools. Using it to its fullest requires a good understanding of TCP/IP protocol. This is a collaborative project that includes a comprehensive range of tools, version tracking as well as meta-packages.

Netsparker Security Scanner: An automated web scanning is aimed to detect different vulnerabilities, such as cross-site scripting and SQL injection. Developers can customize the security scan by using various attack options, authentication, and URL rewriting rules in case Netsparker can scan up to 1000 web applications at once. Netsparker offers a proof-based scanning technology that produces accurate results.

Users are allowed to capture and analyze network packets by a famous network analyzer Wireshark. Wireshark allows the users to capture even small details of network activity such as, source and destination protocol, with the help of live capture and offline analysis tools. Additionally for intuitive analysis, the program also gives us color guidelines which are optional.

Metasploit is the most preferred automation framework for penetration test. It provides an awareness, prepare, and advance function, a manage, and validate role which is directly aimed at professional teams. The application has a capture and command line interfaces and provides test results for more than 1500 begging to every type of user. MetaModules in Metasploit can also be used for network segmentation tests.

BeEF: Sometimes referred as the Browser Exploitation Framework, BeEF is designed to analyze security vulnerabilities away from the client system and network perimeter with most focus on web browser exploitability. The tool interacts with various web browsers to run instructed modules and applies client-side attack vectors to test security posture.

John The Ripper: The Ripper is likely the best known password cracking tool, and passwords are one of the most common points of vulnerability. The password cracking systems of this pen testing tool consists of a particular cracker. It automatically detects different types of password hashes and search password vulnerabilities in the databases.

Aircrack: Since Aircrack NG does capture data packets and exports the data through text files for study, it is designed for detecting security holes in wireless connections. The tool pays attention to some areas of security, such as attacking, monitoring, testing and cracking. It also provides a faster rate of tracking compared with most penetration tools.

Acunetix Scanner: It is an automated harnessing tool that is capable of auditing complex management data and identifying compliance issues. The software is capable of dealing with a

range of network vulnerabilities, for instance Cross-site scripting and SQLi testing, including very sophisticated XSS detection. Acunetix communicates with famous issue trackers and WAFs and covers over 4500 vulnerabilities.

Burp Suite Pen Tester: Developers can get two versions of the program, one of which is free and includes all the essential functions. Burp Suite is a tool that people use for assessing the security of web applications. Functionality of the tool includes interception of traffic between web browser and target application, modification of traffic before sending it, and analysis of responses.

These are the top penetration testing software and tools that are vital for security assessment management, awareness raising, and provision of resources for defenders to always be a step ahead of the challenger.

(Chougul, 2019)

In terms of IoT, the attack vectors are very different from the traditional web applications. Software vulnerabilities attacks web applications while other hardware components like firmware, networks, wireless, cellular, web-based applications, and cloud APIs attacks IoT devices. These attack vectors render IoT devices open to many security threats that can disrupt their performance or even cause harm to users. A holistic approach to the security of IoT should be adopted to intervene all these attack vectors, and devices themselves, and the users they serve to protect them. Through the implementation of strong security measures like encryption, access control, and routine updates, we can reduce the risk of IoT attacks and ensure the security of these devices.

2.3.3 Best Practices for Penetration Testing in IoT Systems

(Engebretson, 2013)

Posit that a successful penetration test involves some key steps. The first element to consider is the reconnaissance which implies collecting all the information that can be found about the target. This is an important stage since it allows to determine the target's vulnerabilities and weaknesses and develop the attack. The following stage is scanning during which port scanning and vulnerability scanning are conducted to identify open ports, possible services, and particular vulnerabilities in the software and services of the target. These two phases will be more detailed in chapter two and three.

Having collected the outcome from scanning stage, the tester proceeds to the exploitation stage, in which the tester employs the information gathered in the first two steps to assault the target. This step is the most difficult and the most thrilling part of the process, as it utilizes diverse techniques, tools, and code to attain administrative access to the target.

Post exploitation and maintaining access are the last activities. After the tester is granted administrative access to the target, they need to establish a more persistent backdoor in the system, so that the access does not disappear if the system reboots or shuts down.

It is important to keep in mind that every stage of the penetration testing process is important, and the success of the process as a whole depends on how well each stage is performed. The subsequent chapters will also cover each stage of the process, the tools, methods, and the codes used.

(Akhilesh et al., 2022) argue that, unlike a testbed, Pentos performs attacks like a password attack, Wi-Fi attacks, and online scanning. Though it outperforms many methods in most aspects, it is limited to a PT over networks since it does not test for OS or software security

weaknesses. Furthermore, the PENTOS is not suitable to test such IoT protocols like MQTT and CoAP and the ZigBee PT module is in the process of development.

Modern tools, a better form of human PT processes are needed to tackle these limitations. Utilization of automated security evaluation methods allowed to significantly enhance security of IoT device while keeping expenditures under control.

Interface Testing: This type of testing focuses on input validation testing and is performed on the interface used for communicating with IoT devices.

Testing of the transport is performed to the IoT device network, cryptography protocols related and message transmission protocols (protected).

Software, firmware, operating systems, and system services are tested through system testing from the purpose of identifying implementation defects, unsafe system configurations, and other known vulnerabilities.

(Chougul, 2019)

The firmware penetration testing is a major step in discovering and eliminating the possible risks of security in gadgets. In this approach, different techniques are put to use which include binary analysis, reverse engineering, file system analysis as well as observation of crucial keys and certificates. These techniques enable to detect the loopholes in the firmware, which are then corrected to make sure the device is secure.

Firmware modification is one of the major aspects of firmware penetration testing, which includes firmware being changed in order to remove any perceived security risks. For example, this can be achieved by deleting the code not needed, adding the code to improve the security, or changing the firmware to stop the unauthorized access.

The hardware should also be made tamper resistant to avoid unauthorized access to the firmware. This can be done through several actions, e.g., the implementation of tamper-resistant seals, introduction of tamper-proofing mechanisms, that the device is built in a way which prevents the access to firmware.

In addition, provision of firmware updates and patches is paramount to keep the device safe. The updates will fix any security problems that have been discovered and that the device is protected against new security issues.

Usage of strong authentication, encryption, secure protocols, and determining the method of delete at the time of device failure is crucial for creating secure firmware. These steps will help organizations to guarantee firmware security as well as safeguard devices they make so that they do not get exploited by attackers.

It is essential to conduct firmware penetration testing so as to ensure that devices are secure and that they do not possess any potential threat of security. The process involves binary analysis, reverse engineering, investigation of different file systems, and checking on essential keys and certificates. Creating secure firmware businesses should also build their hardware to be tamper resistant, provide firmware upgrades and patches, and develop methods to protect data upon device disposal. In doing so, they are able to ensure that they protect their customers and keep their reputation intact in the market.

2.4 Cultural & Human Factors Related to DevOps & IoT

2.4.1 Mindset Shifts for IoT Adoption

DevOps started from the necessity to resolve the contention issue between development and operation teams, as was said by P. Debois at Agile in Toronto in 2008 **(Debois 2008)**.

(Díaz et al., 2021)

DevOps has transformed to a cultural movement that seeks to promote cooperation among software development, deployment, and operation stakeholders, for the purpose of delivering high-quality products or services in little time.

Nevertheless, its straightforwardness, DevOps adoption is challenged by dramatic differences from usual ways of working. Adopting DevOps is a challenge that organizations need to invest a lot of effort in, requiring the support of CEOs, CIOs and practitioners. DASA (the DevOps Agile Skills Association) outlines that main drivers for the adoption of DevOps include ensuring that the IT activities are more convenient, quick and cheap, and that the organization is delivered with more business value. This involves the shortening of time-to-market, speeding up innovation, reducing costs, improving communication and cooperation within teams, reducing errors, and increasing system stability.

Research conducted on DevOps practices in software development organizations had shown that the main reason for the adoption is the long software delivery process. The principles of DevOps propel process automation, efficiency, and teamwork, which leads to quicker time-to-market, better software quality and more satisfied customers.

(Khan et al., 2022)

The challenges that organizations face in implementing DevOps practices stem from the requirement of development and operations skills and the change in mindset. There are various organizations that do not have qualified employees who possess required technical know-how and the understanding of DevOps concepts, methodologies, tools, advantages, and challenges. Besides, there is always a lack of training and interest to study DevOps and employees are interested only in their narrow areas of specialization, which causes many challenges. Practices of criticism, like blaming culture, may ruin DevOps culture, with surveys indicating team

culture as the main resistance to successful adoption. Blaming culture breeds negative behaviors, such as punishments, conflicts, and accusations, which lead to friction in workflow. Resistance to change is yet another major challenge with leaders many times favoring personal interest over organizational goals hence resulting into conflicts and role changes. Trust and confidence problems are common in DevOps producing refusal out of fear of losing the job. According to the study, there are ten main challenges for the culture adaption of DevOps, shortage of collaboration and communication, knowledge and skills gaps, complex infrastructure, management issues, lack of DevOps methodology and trust issues are the most critical challenges. Solving these challenges is crucial for the achievement of DevOps.

(Jha et al., 2023)

In the past, developers and administrators had different goals which resulted in the creation of different features and maintenance, also, outages, blame-games and customer dis-satisfaction. DevOps, derived from the word development and operations, seeks to enhance collaboration and automation to solve these issues. It is about approaches, instruments, and attitude aimed to reduce iteration cycles, support faster feature implementation, and improve interaction between developers and operations teams.

Origin of DevOps can be tracked to the attempt by Andrew Shafer to organize an “Agile Infrastructure” meetup in 2008, which gave rise to DevOps discussion groups and “DevOpsDays” events while the Gartner assessment in 2011 uplifted the adoption of DevOps by organizations, and the State of DevOps reports and advancements in cloud computing and containerization technologies i.e. DevOps is a ever-evolving way of software delivery, which is widely used today, continuously changing to adhere to the needs of the industry.

(Brous et al., 2020)

The influence of IoT adoption on enterprises is predominantly determined by the data it generates. which is "Big", "Open", and "Linked" (BOLD) in nature. Primarily, the Internet of Things (IoT) produces massive amounts of high-level data records that are often declared as having a higher accuracy, heterogeneity, variability, and quantity in comparison to old source of data. While there are difficulties with dealing with this "Big" data, including data management and limited capacity of the existing IT infrastructure, new technologies are making it easier to cope with these problems. Another important aspect of IoT is the openness that allows the data to be reused for various applications and thus creating new insights and opportunities. The IoT development process largely consists of unifying these technologies into one whole – identification and tracking, sensor networks, and communication protocols. Organizational implication is also big, involving more or less alteration on business workflow and it still remains hard to sum up the process by automating it. The process of decision-making in organizations which involves many stakeholders is complicated and lengthy, and it can slow down the process of innovation adoption. The IoT-related population-related changes may preclude people's adaptation to new technologies, the cultural component of which demand Additionally, particular attention to implement this successfully.

Discovering and using positive results, as well as reaching expected benefits are two approaches that are vital in implementing and sustaining IoT adoption within organizations, stressing the key skills of staff for such forms of these staff to carefully observe results and learn replacing poor behaviors with good ones in practice.

2.4.2 Role of Leadership and Training

(Maroukian & Gulliver, 2020a)

When deciding on the adoption of DevOps, different leadership styles among organizations working towards agile, lean, and DevOps practices should be taken into account. Some of such styles include Transactional Leadership, Authentic Leadership, Transformational Leadership, Servant Leadership and Ad Hoc Leadership. Research has revealed that transformational leadership is related to the performance of an organization. Trans formational leaders concentrate on establishing a positive role model, motivating others, promoting new ideas, and considering individual needs. Leaders of successful organizations are usually servant-like, and they support and inspire their teams, creating an atmosphere of trust, cooperation, and mutual help. Ad hoc leadership is additionally an aspect in software development which entails varied groups, which include the team, customer, and management. In this context, the leadership style can be dynamic. Researches stress on the requirement of ongoing leadership in DevOps, rather than for particular projects only. Leaders need to set themselves to the task of serving their teams and creating a working environment where all can work together and grow.

(Maroukian & Gulliver, 2020b)

The ‘State of DevOps’ Report points out an association between transformational leadership and organizational performance. Idealized influence, intellectual stimulation, inspirational motivation and individualized consideration are the means through which transformational leaders inspire and transform followers. Moreover, leader who with a servant leadership approach promote better team performance by establishing an atmosphere of trust, cooperation, and mutual service. Servant Leadership, which was originated by Robert Greenleaf in 1970, is a model that involves followers in relational, ethical, emotional, and spiritual aspects that allow them to develop into their highest potential. Indeed, this style of leadership is based on virtues such as honesty, benevolence, humbleness, compassion, and empowerment. The paper under

consideration is a study which focuses on the features of both transformational and servant leadership by means of a qualitative and quantitative research design. This research paper analyzes the effect of individual leadership versus team leadership roles on team performance in the software product development and coding pipeline fitness. Although there are different schools of thought on whether the leadership role should be individual or team-based, most are united in the belief that a DevOps leadership role is essential.

(Pang et al., 2020)

DevOps understanding and lack of skills are introduced as substantial obstacles by the researchers and the industry practitioners. This should be achieved through determination of the essential DevOps skills, and this can enable practicable implementation of DevOps education. Based on industry perspectives, DevOps job requirements cover a wide set of skills. Such tasks include designing, building, and operating technology stacks, configuring, monitoring, and managing systems, programming tools in diverse languages, scripting Linux/Unix processes, cloud products operations, IT processes automation, best practice enforcement, and possessing good interpersonal communication skills. DevOps practitioners are expected to be proficient in different IT processes and tools and they require the skills of architects, programmers, scripters, system administrators and others. Other important soft skills include management, leadership, and communication. Although there are no strict prerequisites for the DevOps practitioner, experience has a significant influence. Lee recommends that it is ideal for the practitioners to have an IT experience for at least five years. That said, basic DevOps understanding can be acquired through academic training, while the practical experience is developed on the job. Attempts to put up leadership, rules, and norms for DevOps are being carried out by different forms that include writing, communities, conferences, and training courses. The global scale of All Day DevOps, DevOps World - Jenkins World, and DevOps Enterprise Summit among other DevOps conferences and events makes these forums

popular as they are the places for knowledge sharing and networking. Furthermore, DevOps is getting attention in the academic sector as well, as such conferences as the Centers for Advanced Studies Conference (CASCON) and the International Workshop on Continuous Software Evolution and Delivery (CSED) have DevOps workshops. Organizations such as DevOps Research and Assessment (DORA) are doing research and advocating a scientific view on software development and organizational change. The focus of these initiatives is to improve awareness, training and learning in the DevOps area by drawing from information provided by IT practitioners and research.

(Abidi et al., 2023)

The resource-based view philosophy also calls for the need for strategic thinking leading. According to RBV, a company's internal resources and capabilities play a pivotal role in enabling it to achieve a sustainable competitive advantage. A strategic thinking leader is in a perfect position to identify opportunities to align the company's resources to the external environment to make it outstanding in the competition. To do this, the leader needs to have a good understanding of the company's strengths, weaknesses, and opportunities. IoT has the potential to transform every aspect of our lives and, as a result, strategic-minded leaders are needed to help businesses tap into the trend. Such leaders can identify potential possibilities and threats posed by IoT, hence develop plans to leverage the opportunities. With a better understanding of how IoT could disrupt its business models, products and services, and operations, a company can position itself strategically to win. Therefore, in the modern digitally enabled business environment filled with uncertainties and rapid technological change, strategic thinking leadership is crucial. Leaders with strategic abilities can make reasonable decisions and changes for their organizations to remain relevant.

Damanpour, F., & Gopalakrishnan, S. (1998)

The incorporation of innovations within an organization, especially in stable environments like public organizations, is driven by the organizational structures and culture. According to the conclusion of the article provided by author the more rigid the hierarchical or mechanistic layer of an organization, the less often these organizations adopted innovations due to the considerable stability of their environments. On the other hand, organizational forms such as organic and adhocratic are more effective when it comes to discovery and innovation with an environment that is liberal and trusting. In the public sector, organizations are mainly interested in being predictable and transparent, which might make some of them to be hesitant on the adoption of new methods.

Conversely, indeed own managers can exhibit implicit doubt even to well-known systems, which are beyond their knowledge or control, that can be a basic for disrupting IoT. Psychological hesitation about IoT may be one of the factors affecting its acceptance, therefore the issue of trust is crucial for the realization of the system. The trust issues and the culture of trust are the main factors that need to be solved if we want to successfully introduce the IoT within organizations.

2.5 Intersection of IoT and DevOps

2.5.1 Benefit of DevOps Practices in IoT Development and Deployment

(López-Peña et al., 2020)

IT Operations monitoring is, in the opinion of the author, critical for maintaining the reliability and performance of IoT systems. Historically, development and IT operations groups work in the silo mode where they have different goals, processes, tools, and management areas. This lack of working together delays timely feedback for maintenance and updates, hence compromising customer satisfaction and business impact. In such case, DevOps is making its

appearance as a cultural practice aimed to promote tight and more effective cooperation between development and operations teams.

In his case study, the author found that his work is concentrated on the application of DevOps principles in creation of trustworthy IoT systems. His paper formalizes an activity with the name F&CF availability through SPEM 2.0, which is to help the input from operations to development for maintenance and improvements of the IoT systems. The F&CF availability activity enables the DevOps teams get feedback from operations by tracking anomalies or failures in IoT hardware and software infrastructure. The monitoring system and on-demand monitoring components are built by employing Infrastructure as Code (IaC) and Monitoring as Code (MaC) techniques that support versioning, automation, virtualization, and containerization. A case study on a portable water supply system shows that this monitoring infrastructure perceives whether availability is in the expected state and predict anomalies during production deployment. Though the case study is qualitative and has limited external validity, it offers useful information about the actual effectiveness of the applied activity and process practices. Despite the drawbacks, the case study confirms the contributions postulated by the research, providing real life proof of the advantages of embedding DevOps practices in IoT system development and maintenance.

(Guşeilă et al., 2019)

In his studies state that DevOps enables fast delivery of new functionalities and products to end-users via short release cycles, improving customer centricity and flexibility to changing needs. Nevertheless, due to rapidly changing IoT applications requirements, an automated platform is essential to test the hardware and the software. An iterative development process is observed to be the most suitable for IoT systems deployed in cloud environments with distributed components. The study provides implementation approaches, tools, and practices for automation in continuous integration, deployment, and testing in agile software

development lifecycle of IoT applications. A suggested DevOps pattern and agile software stack, created upon opensource solutions, provides a complete software delivery pipeline. Continuous testing, that includes best practices and methods, can be smoothly implemented in any cloud environment no matter which cloud provider is chosen. The study results in the standardization of the software delivery process, identification of solution architecture components for continuous integration and build, and specification of test scenarios and phases for the introduction of continuous testing in a DevOps pipeline. Such results give important clues for simplifying the development and in the future release of IoT applications.

(Aktaş et al., 2023)

Also, certain problems arise from the difficulty of combining different software technologies into a single system in the IoT sphere, since the apparatus to many systems of this nature often takes a long time. Therefore, at the moment, in the development of IT projects for the IoT sphere, DevOps practices are being implemented. In particular, in the course of IoT projects, it is necessary to ensure accurate functioning of coding changes and integration into the system. The project described in the work successfully monitored metrics and alarming, for the implementation of which the methods of monitoring the system in DevOps, suitable for the IoT sphere, were used. As the author notes, the project met its goals as effectively as possible, which demonstrates the full satisfaction of the requirements of the IoT sphere. This library can be further developed to ensure the use of an even greater number of metrics that need to be monitored. In an industrial enterprise, for example, a production line, the system can detect potential issues at an early stage, which will reduce downtime and increase productivity. At the same time, in Home IoT, a warning system has been developed that allows you to maintain the health of the equipment used and, therefore, save money and save on their disposal.

2.5.2 DevOps Challenges in IoT

(Yadav et al., 2018)

Security is the main concern of the Internet, but for the Internet of Things (IoT), it becomes a large-scale challenge. The increasing number of IoT devices, which is forecasted to be measured in dozens of billions, makes it possible to use security vulnerabilities with special focus on bad or low-quality devices. Partial data flow also increases possibilities to data theft, endangering even lives of people. In addition, the copiousness of similar devices collections multiplies the impact of security defects on the whole network. In this regard, privacy is paramount, and the needs go beyond authentic, trustworthy, and confidentiality to selective access and safe communication of business through smart objects. The lack of standards and documentation can lead to disruptive behavior of IoT devices, mostly from low-quality or non-standard devices. Absence of congestion not only pushes the networking resources but also creating disrupters to the products that operate on the internet. Furthermore, the application of IoT requires a highly trained employees to network, hardware, software, and the IoT technology. Nonetheless, in the case of India, the initiative to develop such competencies among the workforce is lacking due to the fear of losing jobs, which poses challenges for organizations during the transition from legacy systems to IoT-enabled systems.

(Bijwe & Shankar, n.d.)

Highlighting the need to recruit staff with a suitable technical knowledge and skills and providing continuous development for the existing employees is important. Nevertheless, the challenges are presented with the fear of change and the uncertainty of how responsibilities would change. Agile software development advantages along with DevOps practices, especially within cross-functional teams in IoT applications, are considerable. The gap of ongoing feedback should be addressed, as well as approaches to testing environments in the IoT applications what are not covered by DevOps technologies. Moreover, lack of visibility

into customer environment during testing environment design continues to compound the complexity of testing and debugging IoT systems.

(Sand, 2016)

Integrating IoT into business and household IT networks presents distinct problems for DevOps that go beyond the typical software development process. These issues encompass rigorous quality assurance and robust back-end support phases. While IoT is primarily driven by consumers, its significance in corporate markets increases. DevOps engineers have the responsibility to guarantee that all advancements in IoT firmware operating systems are traceable and auditable to ensure compliance. Engaging in collaboration with hardware experts and vendors throughout the development process is crucial for the program's reliability and its smooth integration with current IT networks, hence preventing the software application from being tied to a certain vendor. The increasing quantity of IoT endpoints places strain on the global networking infrastructure, highlighting the need to address interoperability, networking, and connection challenges in IoT development, with a specific focus on network environments, protocols, and standards. To provide fast testing and updates for the widespread production and deployment of IoT devices, it is essential to have a well-designed back-end architecture. This architecture should provide full access to the development cycle and a central repository to track all changes made to the devices once they are released.

(Rajapakse et al., 2022)

Security problems deriving from the complexity of tools and integration issues present a large barrier to DevSecOps. Present-day DevOps and security tools are complex especially for the developers that do not have a security background. Lack of proper documentation also contributes to the problem as it does not contain relevant information about security settings, that is, least privilege configurations. The incorporation of testing tools to DevOps pipeline

becomes challenging because of the manual and time-consuming nature of the process. Developers need deep knowledge to build the tools with rights security settings and integrate them securely. Configuration management poses yet another problem as developers hardly adhere to the best practices, and as such, leaving vulnerabilities in the system. Tools using static application security testing (SAST), although critical to early defect detection, cannot be used in rapid deployment cycles which is the effect of a long process of manual assessment. Also, DAST tools demand much effort for their setup and run, therefore, are not much suitable for regular releases. Despite being very popular, container ecosystems are very vulnerable, corrupted images and insecure configurations causing security threats. The CI systems, which are crucial to the DevOps pipeline, become more exposed to vulnerability as other tenants run their code, thus making numerous attack vectors possible. Further, the security constraints of the Continuous Deployment (CD) pipeline provide a way for ransom codes to penetrate the production environment, which has a negative impact. Tools for Infrastructure as Code (IaC) and Configuration as Code (CaC) that are part of DevSecOps are left to insecurity and risks by their developers, which often leads to big losses from breaches. To put it more briefly, these challenges illustrate the requirement to augment the security processes and knowledge across the entire DevSecOps lifecycle to minimize risks efficiently.

2.5.3 Examples Showcasing Successful Integration of IoT and DevOps

(Diaz et al., 2019)

In his paper describes a formalized act in achieving response from Operations to Development in IoT system implementations. This task makes possible the transmission of telemetry from production systems and therefore improves collaboration and transparency among development, IT operations, and security teams. The developers become aware of threat detection from log traces of devices as well as anomaly identification when a self-serve cybersecurity monitoring system is instantiated within a DevOps environment. The case study

illustrated how this monitoring infrastructure was able to detect threats including denial attacks and facilitated predicting spoofing issues. Particularly, the infrastructure setup was done in a DevOps manner which involves automation using scripts and configuration files, version control with GitHub repositories, and automated deployment using virtualization and containerization technology. Despite the fact that only one case study was used to assess the approach's validity, the authors argue that this was enough to validate the contributions made, considering the difficulties in carrying on more than one experiments in software engineering. The plans are to conduct more case studies on industrial applications for further validation of the contributions and expand the activity to Supervision as Code, implement advanced monitoring configurations, and integrate machine learning algorithms for better monitoring and supervision.

(Karapantelakis et al., 2016)

Discusses a system, which supports development and running of the applications for Internet of Things (IoT) that should be connected to mobile networks. The system creates, monitors, and removes applications automatically. Such applications operate with cloud software and mobile devices using cellular network. They ensure that these applications receive a high network quality and speed. The authors created a prototype of such a system and assessed its performance. They concluded that regardless of network traffic, the application’s setup and tear down time was more or less constant. Additionally, while the applications were operating, the system ensured that essential ones got the highest network quality. The researchers also observed that linking to outside locations for additional software increased the time of applications setup. They intend to rectify this later. They also want to be sure the system is able to work with a lot more devices and try it with different applications. Moreover, they want to improve the working of the system by looking at where the data is and how much processing power is close by.

2.6 DevOps and IoT Security

2.6.1 Definition and Principles of DevOps

(de França et al., 2016)

As per author he researched about the DevOps pointed out that there are several definitions of this concept but there is no unison among the researchers and practitioners. Additionally, some of the sources which talked about DevOps did not give a clear definition. DevOps was defined as a framework by a systematic mapping study, but the lack of consensus among other studies was also identified. Dictionaries were used to get an idea of the various concepts related to DevOps. The results indicate that DevOps is mostly connected with the notion of a movement of ICT professionals working together for the enhancement of software delivery by means of a prescribed set of principles, for example, culture, automation, measurement, and sharing. In conclusion, DevOps is an innovation representing the trend of better software development and operations.

As per **(Karamitsos et al., 2020)** DevOps is the most popular methodology, which brings together development, quality assurance, and operations in the sequence of continuous steps. It aims to enable teams to develop high-quality software with greater speed. DevOps isn't a set of tasks, rather it is the culture or attitude that promotes collaboration and cross-functional team communication. DevOps advantage is the fact that it does not need a massive technical change but it to transforms the way teams work in order to be successful.

DevOps principles comprise automation, continuous delivery, and fast feedback. Automation eliminates time waste in repetitive tasks, and continuous delivery is the practice of delivering new features and updates to end-users frequently. Quick feedback response enables teams to react swiftly to issues or user feedback, thereby, increasing the quality of the software.

The pillars which underlie the four DevOps are culture, automation, measurement, and feedback, best practices, and knowledge share. These pillars are abbreviated as CAMS. The ideas “culture” and “technical processes” are similar. Feedback, practices and skillset dissemination fosters teamwork and consistency in the way that automation of routine processes, success measurement and area of potential are undertaken.

Continuous delivery, regular deployments, QA automation, early idea validation, and in-team communication are a few of DevOps practices. Therefore, continuous delivery leads to the end-users receiving new features and updates often and frequent deployments make changes to be applied often. QA automation is for complete testing of software and early detection of problems, while early validation of ideas helps the team to build something that provide value to the end user. Collegiality of the team stresses that all team members are working towards the same ends with free and open communication.

Thus, DevOps is a culture of the team which is directed on collaboration and inter-functional communication. The principles that are core to DevOps include automation, continuous delivery, and fast feedback. CAMS stands for culture, automation, measurement, and sharing of feedbacks, best practices, and knowledge as DevOps pillars. Some DevOps practices that make it possible for teams to use less time in developing high-quality software include continuous delivery, QA automation, frequent deployments, early idea validation, and in-team collaboration.

The study findings suggest that DevOps methodologies for IoT edge computing need to be customized to accommodate the specific demands of edge environments, which are distinguished by novel and distinct challenges. Automation, collaboration, and monitoring were determined to be essential elements for successfully implementing DevOps in edge computing. Additionally, the study suggests that the list of factors affecting DevOps in the edge should encompass organizational and cultural aspects as well. Research has deepened our

comprehension of the difficulties and remedies linked to the implementation of DevOps in edge computing environments. The study has identified the significance of technical, organizational, and cultural factors in the incorporation of DevOps in edge computing. The present study also identified various strategies and technologies that organizations should employ to overcome challenges and ensure effective and dependable DevOps in edge computing..

2.6.2 Role of DevOps in IoT

(López-Peña et al., 2020)

Monitoring the operations of IoT systems is needed for these to be reliable and effective. Nevertheless, development and operations teams normally operate separately which leads to short feedbacks hence, system maintenance and update delays and affect customers' satisfaction. DevOps was designed to promote collaboration and quick feedback between the two teams with continuous feedback from operations to development as a key practice. Nevertheless, continuous monitoring is difficult, especially in IoT systems where high data volumes, network connectivity, and security hazards are some issues. These challenges have been highlighted by previous research, and the DevOps techniques of automation and cross-functional teams can help to control them. Uninterrupted monitoring lessens downtime, improves system performance, and subsequently enhances customer satisfaction.

(Karapantelakis et al., 2016)

DevOps software development focuses on teamwork and integration between developers and operators. Unlike traditional software development methods that separate developers and operators, DevOps attempts to bring the requirements of both teams under the development process so that the development cycle can be more dynamic and quickly adapt to alterations in the operational environment.

The necessity to ensure interoperability between cloud networks, mobile networks, and various device types poses several complex operational requirements, and thus, implementing DevOps for IoT applications becomes challenging. To achieve DevOps in IoT environments, there is a need for a framework that is able to allocate and free network resources dynamically due to metamorphosis of applications into an automatic deployment.

The lifecycle management is part of the DevOps methodology and now even more important in the development of IoT applications. Applications should be redeployed over and over again for DevOps is a retooling process that call for testing done by the development team as well as external use and user input. The issues of automation and quality of service (QoS) in application lifecycle management are important in reducing lead times, enhancing feedback loops, and ensuring quality testing and user input.

(Cowart,2019)

IoT is reshaping the IT industry, and its development is successfully driven through DevOps.

To start with, Spreading Effect stresses the necessity of regular updates of software programs in the servers that are linked to other systems. DevOps makes sure that these updates are done without any problems.

The other effect is titled The Evolution Effect, stating that DevOps will become a critical part of the software development process, as agile development moves towards continuous deployment. The companies that will adopt DevOps and develop multi-disciplinary teams are in line to benefit a lot against their competitors.

Software-Defined ‘Anything’ emphasizes the cruciality of DevOps in IoT. For systems to change their functionality, the software should be updated rather than the hardware being changed physically.

The Infrastructure in Place is a big driver for DevOps in IoT. Given systems connected to the internet and cloud, software can be installed and updated remotely on many devices. DevOps allows the whole process to be done in a very efficient way hence reducing physical interventions.

DevOps encourages Cost Savings and Higher Efficiency. It cuts down on the development cycle by increasing the software quality and at the same time decreasing production costs that result to more productivity and better profits.

Lastly, DevOps enables creation of New Revenue Streams and Business Models. It enables a mechanism of continuous delivery of latest software updates, thus setting up a possibility for offering services that are money making rather than just selling products once.

Landscape of All-in-all, the importance of DevOps in all IoT is crucial and above six reasons are the evidence.

2.6.3 Benefits of Using DevOps for IoT Security

(Ferry & Nguyen, 2019)

Next generation IoT systems require distributed processing with synchronized actions across IoT, Edge, and Cloud infrastructures. The reliability of such systems is mainly based on security and privacy and is thus important for safety-critical applications or fundamental commercial activities. Adaptative nature of their environment and evolving security threats has to be maintained to uphold the dependability and the quality of these systems. Smooth IoT systems deployment over IoT, Edge, and Cloud infrastructures is greatly required.

However, deploying IoT and Edge resources connecting IoT devices to Cloud infrastructures, presents various challenges that require specialized deployment practices meant for the IoT domain. One of the major issues is getting used to a huge heterogeneity, scalability, and

dynamics of IoT systems and their environment. The languages and abstractions needed to organize and provide software services for a multitude of IoT devices, some of which may be only partially or not at all connected to the Internet, are usually absent in the current Cloud and Edge solutions.

Further, DevSecOps philosophy advocates for an integrative approach towards security and privacy in the entire DevOps lifecycle. It propagates the collaboration and sharing of security and privacy-related information throughout the DevOps toolchain. This method also means that security and privacy are made crucial and are not an up on thought but are integrated in the continuous deployment process.

Solving these issues and adopting principles of the DevSecOps philosophy enables IoT systems to be deployed and operated with more reliability, flexibility, and security.

(Griffin, Tatar, & Yankson, 2022)

DevSecOps is al automated-first approach that integrates security at every development stage of software lifecycle. Application of DevSecOps in various systems facilitates risk identification from the design phase to the live stage. This also includes integrating advanced monitoring measures, like threat detection and alerting capabilities. The approach includes a number of security practices including static analysis, source code reviews, vulnerability scanning and checking for known vulnerabilities in third-party components. Furthermore, dynamic analysis and complete penetration testing are performed to reveal possible weaknesses. While there are several DevSecOps models developed for Cloud applications and IoT, there is no literature available for DevSecOps for MQTT.

When implementing a DevSecOps approach, the architecture of an IoT system using the MQTT protocol should be thoroughly threat modelled to identify potential threats. The risk posture for all is determined from that DevSecOps activities and integrated alerting

mechanisms are used to inform all concerned parties about the outcomes. This research paper discusses a comparison of deploying the DevSecOps approach on three unique port services of the Mosquitto, an open-source MQTT broker. The results reveal that our model succeeds in reaching a common level of maturity in security systems what makes us possible to detect and anticipate threats far in advance.

2.6.4 Challenges of Implementing DevOps in IoT Environments

In his study, **(Bijwe & Shankar, 2023)** discovered several hindrances to DevOps applications in IoT. To begin with, there was a lack of competent people, stressing the troubles encountered in recruiting good candidates for DevOps positions. Moreover, the study pointed that the organizational culture has a significant impact on the specification process, indicating, therefore, the requirement for a DevOps culture that is congruent with the objectives of the application.

The study also identified issues connected with the accessibility of required technologies and instruments. Limitation of the use of Agile techniques was discovered and the associated challenges were analyzed. A metamodel proposed, also aimed at IoT designs standardization, and making them adopt continuous integration and continuous deployment (CI/CD) practices.

As far as collaboration is concerned, the study stressed on the need for hiring of staff with relevant technical knowledge and giving them full training. The organization’s resistance to change and inconsistency in the role changes were found to be the barriers to successful DevOps implementation.

Testing and debugging of IoT systems arose as another major issue because of the high number of devices. In this context, the study indicated that DevOps technologies do not cover testing environment for IoT applications, thus, effective methods should be developed. Further on,

restricted access to customer environments during the creation of test environments was recognized as a hurdle.

In conclusion, issues like the unavailability of the right professionals, impact of organizational culture, availability of technologies and tools, lack of collaboration and complexities in system testing in IoT applications. Solving these problems is an important condition of DevOps practices realization in IoT environment.

2.7 Case Studies Related to IoT

2.7.1 Case Studies of IoT Security Testing

(Abdalla & Varol, 2020)

In his research paper conducted a comprehensive study about the security and privacy issues associated with the Intelligent Onvif YY HD IP camera. The study is focused on the discovery and expose of the vulnerabilities of the security components of the camera by using different penetration testing techniques.

The IP camera has a range of functionalities that include two-way audio, pan-tilt-zoom (PTZ) control, and wireless on local area networks. Nevertheless, the study exposes a number of weaknesses and defects in the security system of the camera.

Default credentials are one of the major vulnerabilities that make the camera to be accessed by unauthorized persons. In addition, the default camera identification number can easily be found and used to identify it quickly by potential attackers.

Besides, this research also demonstrates the way of sensitive information transmission that is not encrypted, so the data is open to be intercepted. In some cases, less powerful encryption techniques are used, which makes the transmitted data unsafe.

The author also considers the mobile and Windows applications that accompany the IP camera wherein vulnerabilities such as plaintext storage of sensitive data as well as lack of strong protection mechanisms are found. The vulnerabilities make the applications targets of data breaches and unauthorized access.

Moreover, the examination observes the usage of the IP camera's Real-Time Streaming Protocol (RTSP). It reveals that RTSP operates as an open service with no authentication requirements, allowing unauthorized persons to view the live video stream without having to provide authentication information.

The author highlights the necessity of addressing these security aspects in light of those findings. It suggests that the manufacturers of the devices should make the security features of IP cameras more advanced by using stronger authentication mechanisms, improving the encryption protocols and securing the associated applications. Overcoming these vulnerabilities enables manufactures to secure user's data and improve security of IoT devices in general.

(Xu et al., 2014)

IoT is the system that is ready to change the way people, companies and governments communicate with each other by linking numerous devices to the internet. Considering the fact that IoT devices are going to outstrip in numbers of personal computers as well as the mobile phones by a huge proportion, it becomes imperative to optimize the design of these devices. The author supports the use of Computer-Aided Design (CAD) methods in conjunction with traditional modeling leading to the efficient development of IoT systems.

Energy efficiency and security emerge as the main worries when it comes to IoT devices. CAD tools are suggested as an effective way to overcome these difficulties. The paper deals with hardware security primitives, which are a stable Physical Unclonable Functions (PUFs), and a

digital PUF, which provide secure transfer of information and public key protocols with low latency.

In the end, the paper seeks to lay a ground for the development of CAD techniques that will be specifically targeted to the special requirements of IoT device design.

(Davis et al., 2020)

The author points out that existing vulnerability studies of IoT devices tend to be insufficient in this regard because they are usually provider or device-specific, therefore, devices from less popular vendors are ignored. They conducted their own research and discovered that many types of attacks such as physical, network, software and encryption attacks are feasible on many IoT devices.

Their study of the Common Vulnerabilities and Exposures (CVE) and National Vulnerability Database (NVD) repositories showed that lesser-known vendors' IoT devices were underreported, which may mean that strong security protocols were not in place.

Their experiments imply that popular vendors and devices on the whole have better security stances in comparison with their lesser well-known counter parts. This claim is backed up by their vulnerability detection results presented in Tables IV and V.

The author advises both national vulnerability repositories as well as the research community to expand their scope to IoT devices of the lesser-known vendors. They propose that security requirements should be standardized for all types of IoT devices and that for this purpose, they should be classified according to the utility and dimension of attack.

To sum up, the author plans to perform a more detailed vulnerability investigation in a smart house environment that will be fully equipped and would involve devices from smaller producers in their future studies.

(Ling et al., 2018)

The paper begins to present their perspective on Internet of Things (IoT) system by covering all aspects of the system, starting from the device and ending with the blockchain and controller. They indicate ten primary functions essential for such system, emphasizing the need to protect them as a result of their risk analysis.

Next, they detail their investigation into the vulnerabilities of an IP camera system, uncovering three types of attacks: device scanning, brute force, and device spoofing. Such attacks enable the attacker to gain full control of cameras of one producer. Based on real-world tests, they prove the efficiency of these attacks, with a device spoofing attack that became especially effective, as it managed to get a user's password with a 98% success rate, unrelated to the password's complexity.

The authors suggest that their comprehensive approach to IoT security and privacy could act as a template for the development of secure IoT systems without blocking user privacy. Through focus on vulnerabilities in specific devices, such as, Edimax cameras, and smart plugs, they illustrate how such vulnerabilities can bring about installation of dangerous malware like Mirai. They also employ modeling and simulations to expect the likelihood of Mirai attack propagation.

To sum up, the paper emphasizes the growing need for IoT vendors to improve security of their products. They promote a holistic perspective to IoT security and privacy, aiming at inspiring and directing research on ensuring that IoT systems are protected from the threats arising.

2.7.2 Case Studies of IoT and DevOps

(Ghantous & Gill, 2020)

DevOps is something like a bridge that serves to link two separate sides of doing things – developing and running them. It is a function of a way of calling agile." DevOps specialists now made tools that allow to quickly and easily deploy IoT applications used in smart devices on dozens of cloud services at once. This paper discusses a new approach called DRA, a plan type that works with any company's tools. It aids in deploying the software to a cloud service that utilizes multiple clouds simultaneously but following the DevOps methodology. The paper also describes case study with this strategy and its overall performance. They discovered that it does a great job of deploying IoT applications on cloud services without being too vendor specific. The test results prove that the plan is helpful for the persons who want to put IoT applications on cloud-based services, and it provides them with specific directions. It also implies on some new scopes for future investigation, such as DevOps with drones and robots.

(Jokela, 2019)

The case study was designed to identify problems in the development platform and understand how the implementation of software development methodologies could assist. It identified typical pain points witnessed in other firms with in-system development projects. Several problems were connected with development procedures, and some were practical problems in development and deployment arrangements. There was also the problem of the absence of tooling for the distributed teams. Despite the fact that the study suggested the areas for improvement on the theoretical basis, it did not apply or measure their impact. The lit review also indicated that automation, monitoring, process changes and versioning improvements could enhance productivity and reliability. Automation was positively associated with productivity increase whilst a number of process changes such as introduction of lean time planning proved to be effective. Versioning and dependency management were very important for the quality and reliability function, with suggestions for automated testing and deployment. In general, the research offers some ideas of IoT and DevOps integrated approach areas that

need some improvements but emphasizes the necessity of further implementation and testing to judge their efficiency thoroughly.

(Thompson, 2019)

The study opened with the statement that security, trust, and privacy are major issues for IoT systems and that they must be handled appropriately. Due to such things as costs and time limitations, the majority of developers forget about these dangers. The research unveiled the present extent of the assistance in the IoT systems trustworthy and secure utilized through DevOps practices is insufficient. So, the study developed a way and a tool of which a developer can use for better risk planning and assessment of IoT system. This method and tool were used in a simulation of a smart home scenario that demonstrated how they would work in practice. They also shared some visions of how to modify the tool as to turn it into better for real-time monitoring or to upgrade it with new options. Nevertheless, they said that as the study was mainly conducted by the author, further tests need to be performed to ensure that the method and tool are suitable for other people as well. They said some parts of their approach have been tested in real industrial settings, which adds some assurance, but they still need to thoroughly test their whole approach in real DevOps situations to see if it's practical and useful.

Chapter 3 : RESEARCH METHODOLOGY

3.1 Research Question

The study's aim to understand the various factor that affects the adoption of IoT System in a DevOps enabled environment. To fulfil the study's objective, the following research

1. What is the impact of the exiting support systems and resources within the organization on the adoption of the IoT systems in DevOps environment?

2. What do peers and leaders' opinions and behaviors influence on the IoT technology adoption decision in DevOps environment?

3. How do expected advantages and perceived demands influence organizational decisions to adopt IoT within DevOps frameworks?

4. What are the primary security issues related to the integration of IoT systems into DevOps environments, and how do organizational stakeholders recognize these issues?

5. What is the impact of perceived risks on making process in a context of IoT technologies adoption in DevOps-enabled environment?

3.2 Research Design

3.2.1 Operationalization of Theoretical Constructs

A theoretical model has been created to examine the factors that affect the adoption of IoT systems in a DevOps enabled environment, drawing on the principles of two theories: the "Technology Threat Avoidance Model (TTAT) (Liang & Xue, 2009)" and the "Unified Theory of Acceptance and Use of Technology (UTAUT) (Venkatesh et al., 2003)". The variables derived from UTAUT are "Effort Expectancy, Performance Expectancy, Social Influence, and Facilitating Conditions". The characteristics extracted from TTAT are perceived threat,

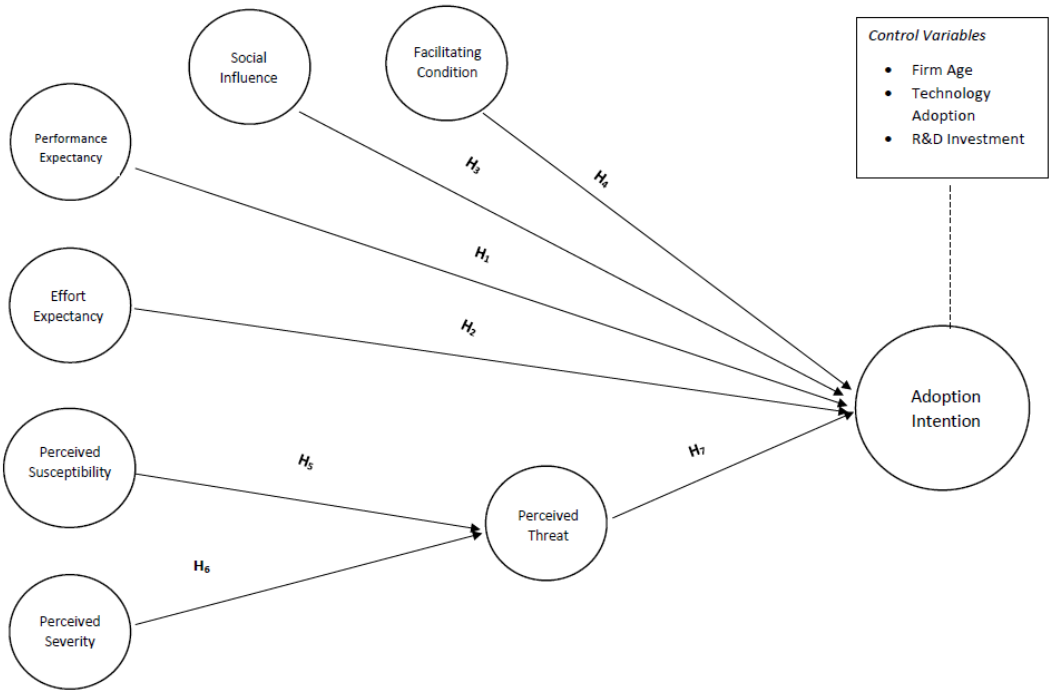
perceived severity, and perceived susceptibility. The UTAUT component of the model, as suggested by Venkatesh et al. (2003), suggests that “Performance Expectancy (PE)”, “Effort Expectancy (EE)”, “Social Influence (SI)”, and “Facilitating Conditions (FC)” are crucial factors that determine the acceptance and usage of technology (Venkatesh et al., 2003).

- Performance Expectancy is the degree to which an individual expects that using a particular system would enhance their job performance (Venkatesh et al., 2003).
- EE refers to the level of usability and user-friendliness of a system, as described by Venkatesh et al. (2003).
- Social influence (SI) is the measure of an individual's perception of the expectations from important individuals over their use of the new system (Venkatesh et al., 2003).
- FC stands for the degree to which an individual perceives a solid organizational and technological framework to be in place to facilitate the usage of the system (Venkatesh et al., 2003).

However, the TTAT component, which was presented by Liang and Xue (2009), focuses on the conduct of avoiding technology adoption. In this component, Perceived Threat (PT) and its susceptibility (PSS) and severity (PSE) are important factors.

- PSS stands for the subjective evaluation of the likelihood of experiencing harm and an individual's perception of their own susceptibility (Liang & Xue, 2009).
- PSE stands for the emotional and moral implications of utilizing the technology, as well as the legal ramifications (Liang & Xue, 2009).
- PT is the measure of worry or anxiety that a person feels about the potential negative outcomes that may result from utilizing a system, such as the possibility of data breaches or other security problems (Liang & Xue, 2009).

Figure 3.1.1 Theoretical Model Used in this Study



3.2.3 The Study's Hypotheses

H_1 – Performance Expectancy will have significant influence on the adoption intention of IoT System in DevOps enabled Environment.

H_2 – Effort Expectancy will have significant influence on the adoption intention of IoT System in DevOps enabled Environment.

H_3 – Social Influence will have significant influence on the adoption intention of IoT System in DevOps enabled Environment.

H_4 – Facilitating Conditions will have significant influence on the adoption intention of IoT System in DevOps enabled Environment.

H_5 – Perceived Susceptibility will have significant influence on the perceived threat on the adoption of IoT System in DevOps enabled Environment.

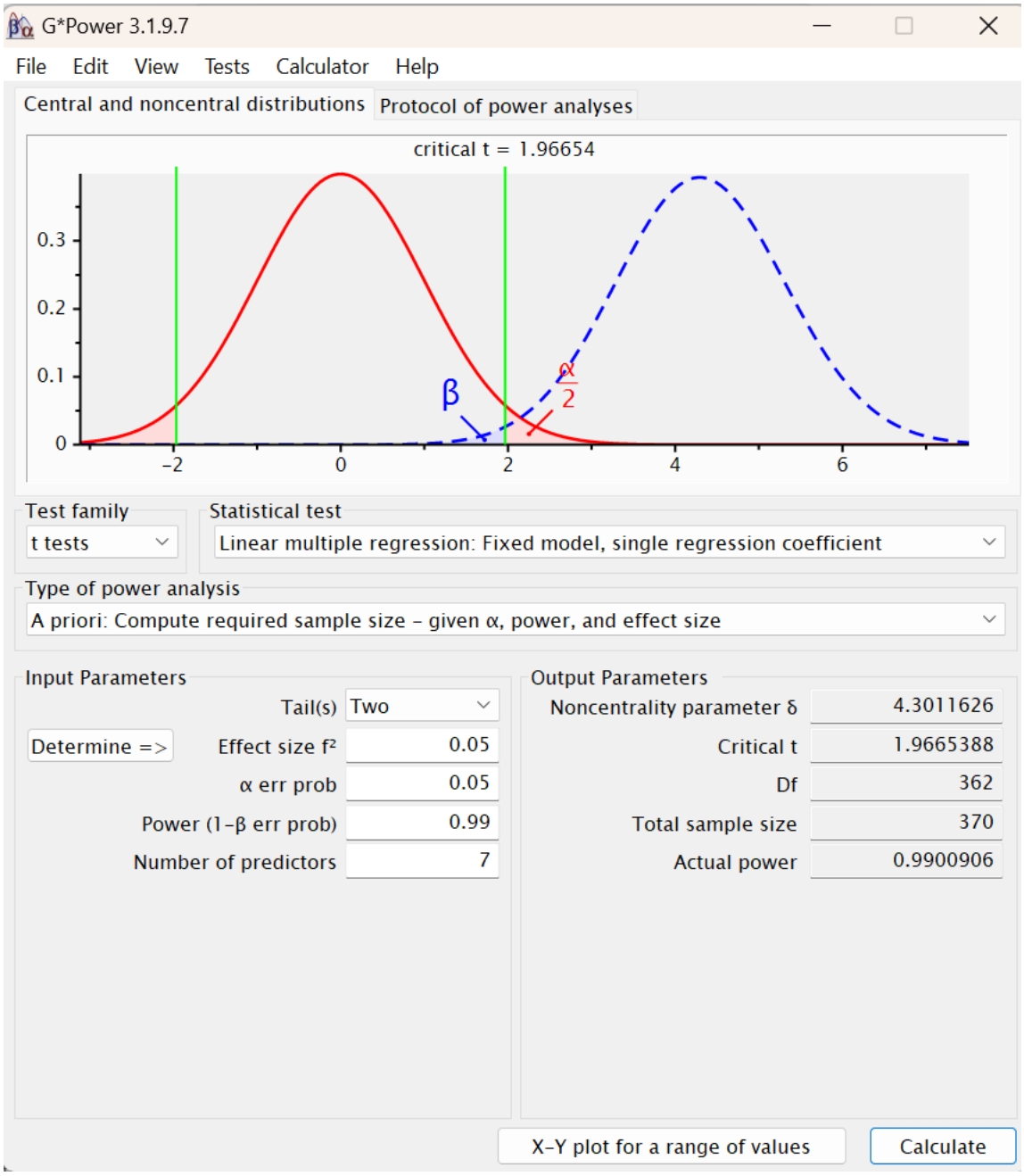
H_6 – Perceived Severity will have significant influence on the perceived threat on the adoption of IoT System in DevOps enabled Environment.

H_7 – Perceived threat will have significant influence on the adoption intention of IoT System in DevOps enabled Environment.

3.2.4 Sample Size for the Study

The G* Power software was utilized to calculate the necessary sample size for the suggested study model. The software's findings are displayed in Figure 3.1.2. To ensure the statistical accuracy of the model and minimize the likelihood of Type I and II errors, the sample size has been set at 450, which exceeds the required sample size of 370. The larger sample size is expected to enhance the reliability of the findings.

Figure 3.1.2 Smallest Possible Sample Size



3.2.5 Sampling Method or Technique

The purposive sampling technique, selected for this study, is well-suited due to the specialized nature of the subject matter related to Internet of Things (IoT) systems in DevOps contexts. This approach enables deliberate participant selection based on their possession of specialized knowledge and experience that are pertinent to IoT and DevOps. This ensures that the collected data is not only relevant but also provides valuable insights.

Purposive sampling is advantageous in scenarios where the quality of the data is more critical than generalizability, such as in studies requiring a deep understanding of complex issues. This technique enables researchers to target individuals who are not only familiar with but are also actively engaged in the integration, management, or decision-making processes related to IoT within DevOps frameworks. These participants are likely to provide rich, informed perspectives that can highlight nuanced factors influencing IoT adoption, which might not be apparent when using more generalized sampling methods.

In practical terms, respondents could include IT professionals, project managers, system developers, and organizational leaders who have direct involvement with or oversight over DevOps and IoT implementations. Their insights will be valuable in understanding the technical challenges, cultural adjustments, and security considerations specific to IoT and DevOps. Additionally, their responses can illuminate how theoretical constructs such as effort expectancy or perceived threat translate into real-world organizational dynamics and technology adoption decisions.

By employing purposive sampling, the study aims to gather data that are deeply rooted in the professional experiences and operational realities of those at the forefront of employing IoT technologies in DevOps contexts. This approach not only enhances the relevance and depth of the findings but also aligns with the study's goal of developing actionable strategies and best practices for integrating IoT systems into DevOps practices effectively.

3.2.6 Data

The primary approach of data collection in this project will involve utilizing a meticulously designed and pre-validated questionnaire., which is designed to gather comprehensive insights directly from respondents who have firsthand experience with IoT systems in DevOps environments. This approach ensures that the data are specifically tailored to tackle the research

questions posed, focusing on the determinants impacting the acceptance of IoT technology within these specialized settings.

The questionnaire will be carefully crafted to incorporate a diverse range of question formats, including Likert-scale questions, multiple-choice options, and open-ended responses. This will allow for a nuanced exploration of both quantitative and qualitative aspects of IoT adoption. The Likert-scale and multiple-choice questions aim to quantify the degrees of agreement or frequency related to specific factors, such as perceived ease of integration, performance benefits, and security concerns. Meanwhile, open-ended questions will provide respondents the opportunity to elaborate on their experiences, challenges, and perceptions, offering deeper insights into the complexities of integrating IoT with DevOps practices.

Prior to its deployment, the questionnaire will undergo a rigorous pre-testing phase involving a small subset of the target population. This pre-testing is crucial for ensuring that the questions are clear, unambiguous, and effectively designed to elicit relevant and meaningful responses. It also helps in identifying any biases or misunderstandings that might skew the data, thereby refining the questionnaire to better suit the study's needs.

The data collected through this structured questionnaire will be primarily qualitative, providing detailed descriptions and explanations that enrich the understanding of the adoption process. This rich dataset will be instrumental in drawing meaningful conclusions and crafting well-informed recommendations for organizations looking to integrate IoT within their DevOps frameworks.

By focusing on primary data, the study gains the advantage of direct relevance and specificity to the topic at hand, offering fresh insights into an evolving field that secondary data might not fully capture. This approach will offer a strong foundation for studying the influence of

different factors on IoT adoption and for formulating strategies to tackle identified problems and opportunities.

3.3. Measurement Scale

The theoretical model proposed for the study consist of 10 constructs. A 7-point rating scale is used to measure the opinion of the respondents with respect to the study constructs. The measurement scale developed on the basis of prior studies are presented in Table 3.1.

Table 3.3. 1 Study Constructs and Indicators

(From 1 – strong disagreement to 7 – strong agreement)

Construct	Indicator	1	2	3	4	5	6	7
Performance Expectancy *(Venkatesh et al., 2012)*	PE01 - I expect that using IoT systems in our DevOps practices will significantly enhance our operational efficiency							
	PE02 - Integrating IoT systems into our workflow will improve our ability to monitor and manage operations in real time							
	PEO3 - The adoption of IoT systems will lead to better quality outputs in our projects							
Effort Expectancy *(Venkatesh et al., 2012)*	EE01 – I believe that integration IoT systems into our DevOps practices would be easy for us to manage							
	EE02 – I feel confident that I can effectively use IoT technologies with minimal effort							

	EE03 – Learning to operate IoT systems within our DevOps environment requires little effort from our team							
Facilitating Condition *(Venkatesh et al., 2012)*	FC01 - I possess the requisite resources to utilize and implement IoT systems							
	FC02 – My team and I have the requisite expertise to utilize and implement IoT systems							
	FC03 – My company/business unit promotes the utilization of IoT systems through a range of supporting efforts.							
Social Influence *(Venkatesh et al., 2012)*	SI01 - My influential peers believe that I should utilize Internet of Things (IoT) systems.							
	SI02 – My colleagues that utilize IoT systems show a more positive attitude towards the utilization of IoT systems in their profession.							
	SI03 - Individuals who hold significance in my life believe that I should utilize Internet of Things (IoT) systems.							
Perceived Susceptibility *(Liang & Xue, 2009)*	PS01 – IoT systems have a significant likelihood of causing security loophole or breaches.							
	PSO2 – It is likely that the utilization of IoT systems can result in misleading information.							

	PSO3 – It is possible that IoT systems may not be able to adequately serve clients.							
	PS04 - The utilization of IoT technology could jeopardize the business's reputation.							
Perceived Severity *(Liang & Xue, 2009)*	PSE01 – The effects of a security breach in an IoT system would be serious.							
	PSE02 – misleading information via IoT- systems could potentially have significant consequences for my employment							
	PSE03 – The failure of IoT systems to adequately serve clients might have serious consequences for the firm.							
Perceived Threat *(Liang & Xue, 2009)*	PT01 – I am concerned that IoT devices could increase the risk of my job stability.							
	PT02 – I have concerns regarding the possible risks that IoT devices can pose to our current systems.							
	PT03 – I view the implementation of IoT systems as a potential risk to the level of service quality.							
Adoption Intention *(Venkatesh et al., 2012)*	IU01 – If given the chance, I intend to utilize IoT systems in my assignments.							
	IU02 – I am open to incorporating Internet of Things							

	(IoT) systems into my current process.							
	IU03 – I can foresee integrating IoT technology as a permanent tool for my position.							

3.4 Data Analysis

The study will utilize Partial Least Squares Structural Equation Modelling (PLS-SEM) for data analysis, using SMART PLS software. a powerful statistical tool designed to handle complex models and multiple constructs. PLS-SEM is especially useful in exploratory studies when the primary goal is theory building or when the research model involves multiple latent variables and their relationships. This makes it an excellent choice for analyzing the multifaceted interactions between the factors influencing IoT adoption in DevOps environments as hypothesized in our theoretical framework.

PLS-SEM stands out due to its ability to handle complex relationships without requiring a large sample size, unlike covariance-based SEM. This characteristic is especially beneficial in a purposive sampling context where the sample might not be large but is highly specialized. Additionally, PLS-SEM is less stringent about data distribution requirements, making it appropriate for real-world data which often does not follow normal distribution.

Using SMART PLS software for this analysis enables the efficient handling of the model's complexities through an intuitive interface and robust computational capabilities. The software facilitates the modelling of relationships between observed and latent variables, estimation of path coefficients, and evaluation of construct reliability and validity. It provides visual representations of the model, which helps in interpreting the relationships and understanding the direct and indirect impact within the variables.

The analysis process will include several steps:

1. **Model Specification** Establishing the structural model with latent variables according to the research questions and theoretical framework.

2. **Measurement Model Assessment**: This entails evaluating the constructs' validity and reliability using a variety of metrics, such as composite reliability, Cronbach's alpha, convergent validity (as determined by Average Variance Extracted), and discriminant validity (as determined by the Fornell-Larcker criterion and cross-loadings)..

3. **Structural Model Assessment**: Employing a variety of indicators, such as Cronbach's alpha, composite reliability, average variance extracted for convergent validity, and cross-loadings and the Fornell-Larcker criterion for discriminant validity, to evaluate the constructs' validity and reliability.

The outcome of this analysis will provide a robust quantitative foundation for understanding how various factors, as delineated by our theoretical constructs, influence the adoption of IoT in DevOps environments. This approach not only validates the theoretical framework but also identifies key drivers and barriers to IoT adoption, providing actionable insights that can guide strategic decision-making in organizational contexts.

Chapter 4 : RESEARCH RESULTS AND ANALYSIS

To fulfill the study's aims, data was gathered from 450 participants with a meticulously designed and pre-tested questionnaire. Purposive sampling was employed for this study due to the necessity of respondents possessing a specific level of expertise as well as understanding regarding IoT and DevOps environment in order to answer the questions.

4.1 Demographics

The study's respondents, as mentioned in Table 4.1.1, represent a wide range of participants from different geographical locations and characteristics. Out of the 450 participants, the bulk are from Chennai, accounting for 44% of the total. Subsequently, Mumbai holds the largest share at 28%, with Delhi, Kolkata, and Bangalore accounting for 11%, 13%, and 4% respectively. Regarding gender, the majority of the cohort consists of males, accounting for 81% of the respondents, while females make up a smaller portion at 19%.

After conducting a more thorough analysis of the age distribution, it becomes clear that there is a notable concentration of individuals who are mature and possess considerable job experience within the 30–40-year age range. Specifically, 48% of the respondents fell within this age group. The 18-30 age group represents a small 7.5% of the population, whereas the 40-50 age group accounts for a significant 39%. The sample primarily consists of a younger population, with individuals above the age of 50 accounting for only 5.5% of the total.

According to the age of the enterprises that the respondents are connected to, the data suggests a preference for more recent establishments. Specifically, Seventy-five percent of the companies have been in existence for less than 20 years. among contrast, the rest 25% of the organizations are operating for over twenty years, indicating a combination of well-established and rising businesses among the group of respondents.

When focusing on the adoption of technology, the findings reveal a strong inclination towards fast integration, as 95% of the participants classify their adoption speed as 'Quick'. Merely 5% of the questioned group perceive their approach as 'Deliberate', indicating a strong focus on adaptability and quick response to technology advancements.

Within the domain of research and development investment, there exists a significant division. Out of all the respondents, 39% are affiliated with companies that have a significant investment in research and development, indicating a clear focus on innovation and progress. Nevertheless, a significant majority of 61% indicate a lack of investment in research and development (R&D), underscoring a potential disparity between the acknowledgment of R&D's significance and its practical execution.

This demographic profile provides insights into the diverse attributes of the participants, with a significant presence from Chennai and Mumbai, a primarily youthful and male demography, and a strong predisposition towards rapid adoption of technology in relatively new companies. The disparity in levels of investment in research and development indicates different strategic priorities and the possibility of gaining detailed insights into the rolling out of Internet of Things (IoT) in a DevOps environment.

Table 4.1. 1 The Demographic Details of those Surveyed

Place		Gender		Age		Firm Age		Technology Adoption		R&D Investment	
Chennai	196 (44)	**Male**	364 (81)	**18-30 years**	34 (7.5)	**Greater than 20 years**	112 (25)	**Deliberate**	22 (05)	**High**	175 (39)
Mumbai	126 (28)	**Female**	86 (19)	**30-40 years**	216 (48)	**Less than 20 years**	338 (75)	**Quick**	428 (95)	**Low**	275 (61)
Delhi	50 (11)			**40-50 years**	175 (39)						
Kolkata	60 (13)			**Above 50 years**	25 (5.5)						
Bangalore	18 (4)										
Total	**450 (100)**	**Total**	**450 (100)**	**Total**	**450 (100)**	**Total**	**450 (100)**	**Total**	**450 (100)**	**Total**	**450 (100)**

Source: *Raw, original information collected directly from the source.*

Note: The numbers in parenthesis represent the percentage relative to the total.

4.2 Results of PLS-SEM

4.2.1 Evalutation of Measurement Models

The measurement models have been evaluated according to the guidelines outlined by Hair et al. (2019) for reporting PLS-SEM findings. The indicator variables used in this study have a reflecting nature. Examining reflective measurement models entails examining their internal reliability, internal consistency, convergent validity, and discriminant validity.

Internal reliability is established by examining the indicator loadings, as presented in Table 4.2.1.

Table 4.2. 1 Data of Indicator Loadings

Construct	Item	Loading
Performance Expectancy	PE01	0.924
	PE02	0.929
	PE03	0.897
Effort Expectancy	EE01	0.913
	EE02	0.912
	EE03	0.897
Facilitating Condition	FC01	0.798
	FC02	0.869
	FC03	0.822
Social Influence	SI01	0.868
	SI02	0.884
	SI03	0.872
Perceived Severity	PSE01	0.928

	PSE02	0.918
	PSE03	0.909
Perceived Susceptibility	PS01	0.798
	PS02	0832
	PS03	0.9
	PS04	0.812
Perceived Threat	PT01	0.817
	PT02	0.816
	PT03	0.802
Intention to Use	IU01	0.923
	IU02	0.853
	IU03	0.897

Source: *Raw, original information collected directly from the source*
Note: PLS-SEM analysis is conducted utilizing the SMART PLS program.

Indicator loadings quantify the extent to which individual variables and their linked construct share variation. Indicator loadings are used to assess the trustworthiness of reflective measuring methods. The data shown in Table 4.2.1 provides a clear representation of the loadings for all the indicators in our measurement models exceed the required critical value of 0.708, as stated by Hair et. al (2019). The crucial value of 0.708 indicates that the linked concept cccounts for over 50% of the variance in the relevant indicator, demonstrating sufficient item dependability. Therefore, we can conclude that this model exhibits a satisfactory degree of indication reliability.

Once the reliability of the indicator has been confirmed, the subsequent phase involves evaluating the internal consistency and convergent validity. The composite reliability and ρA

are utilized to evaluate the internal consistency of reflective constructs, while AVE (Average Variance Extracted) is employed to measure the convergent validity of reflective constructs. The composite reliability, ρA, and average variance extracted (AVE) of our assessment model are displayed in Table 4.2.2.

Table 4.2.2 shows that both the composite reliability and ρA fall within the required range of 0.70 to 0.95. Furthermore, All of the AVE values above the recommended critical threshold of 0.5. Thus, we might infer that our reflective assessment model demonstrates a satisfactory degree of internal consistency and convergent validity.

Table 4.2. 2 Reliability and Validity

Constructs	ρA	Composite Reliability	Average Variance Extracted
EE	0.894	0.933	0.822
FC	0.777	0.869	0.688
IU	0.872	0.921	0.794
PSE	0.891	0.885	0.657
PS	0.743	0.853	0.658
PT	0.913	0.942	0.843
PE	0.906	0.941	0.840
SI	0.871	0.907	0.764

Source: *Raw, original information collected directly from the source*

Note: PLS-SEM analysis is conducted utilizing the SMART PLS program, EE = "Effort Expectancy", FC = "Facilitating Conditions", IU = "Intention to Use", PSE = " Perceived Severity", PS = "Perceived Susceptibility", PT = "Perceived Threat", PE = "Performance Expectancy", SI = "Social Influence".

The last stage in evaluating the reflective measurement model involves verifying discriminant validity, which measures the degree to which one concept is distinct from other constructs in empirical terms. The HTMT (Heterotrait-Monotrait) ratio is employed to evaluate the discriminant validity of the model. The HTMT values are displayed in Table 4.2.3.

HTMT is a measure that calculates the average correlation value across items from distinct constructions, compared to the geometric mean of the average correlations for items measuring the same construct. High values of HTMT suggest low discriminant validity. The data presented in Table 4.2.3 indicates that all the HTMT values of our reflecting measurement model are significantly lower than the cautious threshold limit of 0.85. Hence, it may be inferred that the discriminant validity of our model has been sufficiently proven.

Table 4.2. 3 Hetrotrait-Monotrait (HTMT) Ratio of Correlations

	EE	FC	IU	PSE	PS	PT	PE	
FC	0.780							
IU	0.586	0.692						
PSE	0.272	0.324	0.262					
PS	0.091	0.065	0.269	0.577				
PT	0.354	0.395	0.211	0.777	0.679			
PE	0.563	0.679	0.571	0.356	0.127	0.509		
SI	0.541	0.696	0.637	0.241	0.128	0.208	0.506	

Source: *Raw, original information collected directly from the source*

Note: PLS-SEM analysis is conducted utilizing the SMART PLS program, EE = "Effort Expectancy", FC = "Facilitating Conditions", IU = "Intention to Use", PSE = " Perceived Severity", PS = "Perceived Susceptibility", PT = "Perceived Threat", PE = "Performance Expectancy", SI = "Social Influence".

4.2.2 Evaluation of the Structural Model

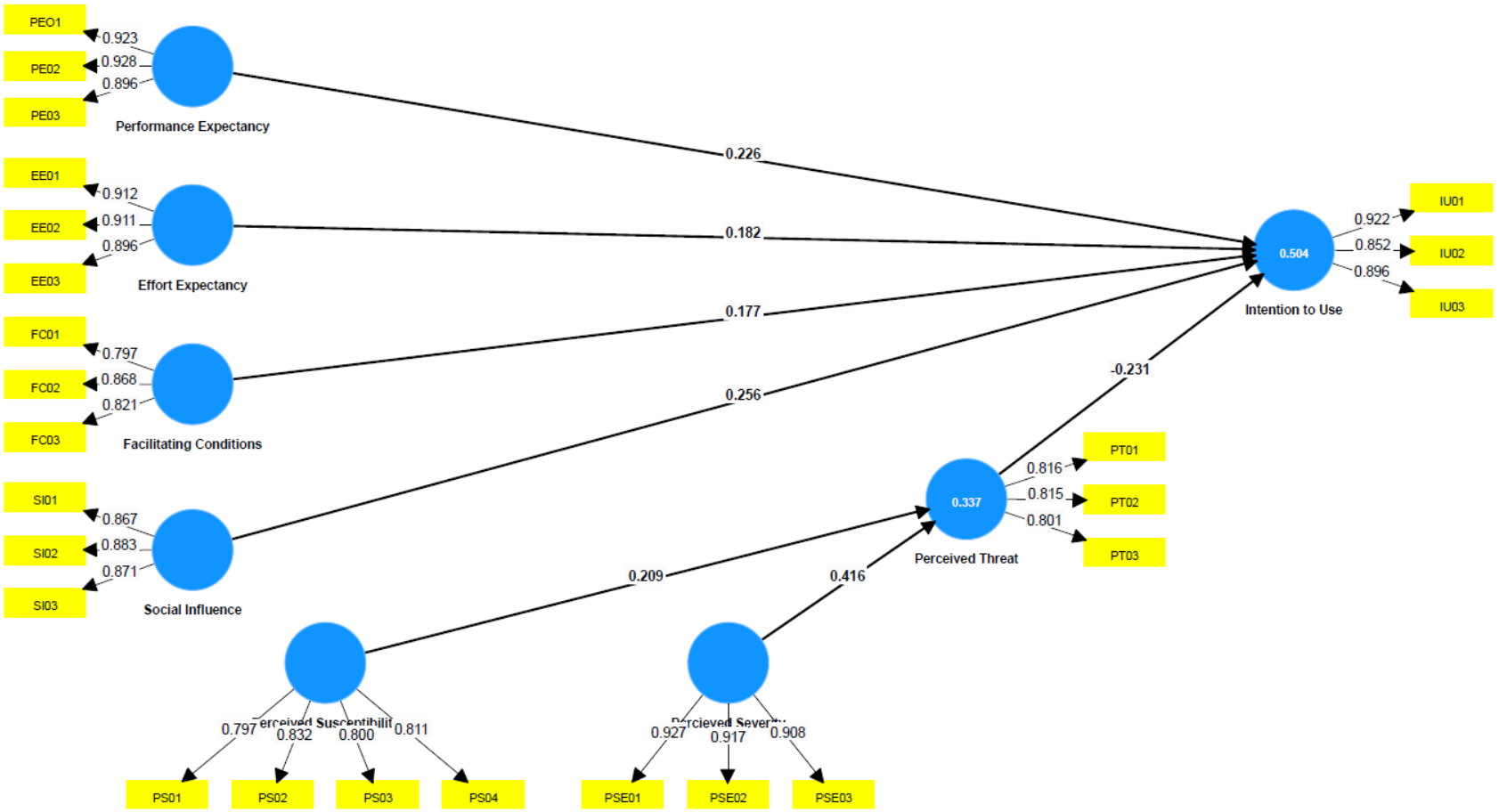
The structural model has been evaluated according to the guidelines outlined by Hair et al. (2019). As stated by Hair et. al (2019), “assessment of the structural model involves three important things viz., checking the collineartity issues, checking the relevance and significance of path coefficientts and checking the models’ explanatorty and predictive power”. The findings of our structural model were presented in Table 4.2.4. The statistical significance of the path coefficients, along with the corresponding hypotheses, was displayed in Figure 4.2.1.

The Variance Inflation Factor (VIF) is used to evaluate the existence of collinearity issues in the model. Table 4.2.4 reveals that the VIF values are around 3. The maximum value of the Variance Inflation Factor (VIF) in our model is 3.625, as reported by Hair et al. (2019). Therefore, it can be concluded that collinearity in the inner model is not at a critical level and will not have an impact on the regression findings. Next, we analyze the magnitude and statistical importance of the path coefficients.

Figure 4.2.1 depicts the magnitude and importance of the path coefficients connecting the endogenous and exogenous factors. The data shown in figure 4.2.1 clearly demonstrates that there is a strong positive connection between perceived susceptibility ($\beta = 0.203$) and perceived severity ($\beta = 0.420$) with the perceived threat. Additionally, there is a positive and significant correlation between performance expectancy ($\beta = 0.226$), effort expectancy ($\beta = 0.180$), facilitating condition ($\beta = 0.180$), and social influence ($\beta = 0.225$) with the intention to use. On the other hand, The variable "perceived threat" ($\beta = -0.231$) is significantly negatively correlated with the intention to use, which is an endogenous construct.An examination of the R2 values in Table 4.2.3 reveals that perceived susceptibility and perceived severity are the key predictor variables in explaining perceived threat (R2 = 0.337). Additionally, perceived threat, performance expectancy, effort expectancy, social influence, and facilitating conditions

are the significant predictor variables in explaining the intention to use (0.504). The model has achieved a satisfactory to high level of success in describing the intention to deploy IoT in a DevOps setting, as evidenced by the R2 value of the endogenous construct ranging from 0.50 to 0.75 (Hair et al., 2019).

Figure 4.2.1 Findings from the Structural Model



Source: *Raw, original information collected directly from the source*
Note: PLS-SEM analysis is conducted utilizing the SMART PLS program

Table 4.2. 4 Results of Structural Model

Outcome	R Sq.	Predictor	Direct Paths & Hypotheses	β	CI	Significance?	f^2	VIF
PE		CV	Firm Age -> PE	0.057	[-0.139; 0.249]	NO	0.007	3.626
		CV	R&D Investment -> PE	0.014	[-0.152; 0.183]	NO	0.025	2.658
		CV	Technology Adoption -> PE	-0.098	[-0.437; 0.238]	NO	0.195	2.172
EE		CV	Firm Age -> EE	-0.042	[-0.227; 0.147]	NO	0.005	3.626
		CV	R&D Investment -> EE	-0.139	[-0.4; 0.026]	NO	0.015	2.658
		CV	Technology Adoption -> EE	-0.037	[-0.356; 0.272]	No	0.028	2.172

PS	CV	Firm Age -> PS	0.114	[-0.066; 0.279]	NO	0.004	3.626
	CV	R&D Investment -> PS	0.003	[-0.174; 0.171]	NO	0.007	2.658
	CV	Technology Adoption -> PS	0.09	[-0.283; 0.406]	NO	0.02	2.172
PSE	CV	Firm Age -> PSE	-0.012	[-0.198; 0.173]	NO	0	3.626
	CV	R&D Investment -> PSE	0.003	[-0.174; 0.171]	NO	0.012	2.658
	CV	Technology Adoption -> PSE	-0.077	[-0.568; 0.366]	NO	0.006	2.172
SI	CV	Firm Age -> SI	-0.078	[-0.263; 0.106]	NO	0.008	3.626
	CV	R&D Investment -> SI	-0.178	[-0.349; 0.102]	NO	0	2.658

		CV	Technology Adoption -> SI	0.159	[-0.112; 0.437]	NO	0.023	2.172
FC		CV	Firm Age -> FC	-0.084	[-0.278; 0.109]	NO	0	3.626
		CV	R&D Investment -> FC	0.022	[-0.142; 0.182]	NO	0.007	2.658
		CV	Technology Adoption -> FC	0.148	[-0.171; 0.464]	NO	0.043	2.172
PT	0.337	PS	Perceived Susceptibility -> PT	0.204	[0.048; 0.372]	YES	0.05	1.025
		PSE	Perceived Severity -> PT	0.421	[0.227; 0.594]	YES	0.212	1.07
		CV	Firm Age -> PT	0.127	[-0.209; 0.448]	NO	0.015	3.636
		CV	R&D Investment -> PT	-0.035	[-0.156; 0.095]	NO	0.03	2.704

		CV	Technology Adoption -> PT	0.007	[-0.262; 0.251]	NO	0.025	2.205
IU	0.504	PE	Performance Expectancy -> IU	0.256	[0.145; 0.365]	YES	0.056	1.413
		EE	Effort Expectancy -> IU	0.181	[0.087; 0.282]	YES	0.018	1.937
		SI	Social Influence -> IU	0.256	[0.145; 0.365]	YES	0.018	1.496
		FC	Facilitating Condition -> IU	0.181	[0.064; 0.291]	YES	0.043	1.424
		PT	Perceived Threat -> IU	-0.232	[-0.315; -0.152]	YES	0.166	1.174
		CV	Firm Age -> IU	-0.015	[-0.147; 0.124]	NO	0.001	3.795
		CV	R&D Investment -> IU	-0.013	[-0.138; 0.112]	NO	0.001	2.839

	CV	Technology Adoption -> IU	-0.188	[-0.505; 0.118]	NO	0	2.633

Source: *Primary Data*

Note: PLS-SEM analysis is conducted utilizing the SMART PLS program, EE = "Effort Expectancy", FC = "Facilitating Conditions", IU = "Intention to Use", PSE = " Perceived Severity", PS = "Perceived Susceptibility", PT = "Perceived Threat", PE = "Performance Expectancy", SI = "Social Influence", CI = "95% bootstrap two-tailed confidence interval", CV = "Control Variable", SI = "Social Influence".

4.2.3 Analysis of Mediation

The importance and power of the mediating constructs have been evaluated using a bootstrapping technique, which provides a 95% confidence interval. The findings are displayed in Table 4.2.5.

Table 4.2. 5 Data of Structural Mediation

Path	β	CI	Significance?
PS -> PT -> IU	-0.066	[-0.117; -0.026]	Yes
PSE -> PT -> IU	-0.153	[-0.206; -0.098]	Yes

Source: *Raw, original information collected directly from the source*

Note: *PLS-SEM analysis is conducted utilizing the SMART PLS program, IU = "Intention to Use", PSE = " Perceived Severity", PS = "Perceived Susceptibility", PT = "Perceived Threat".*

The chart clearly demonstrates that perceived susceptibility (β = -0.066) and perceived severity (β = -0.153) show a substantial negative impact on intention through perceived threat.

4.3.4 Relevance of the Model Prediction

The data presented in Table 4.2.4 indicate that the model has demonstrated a satisfactory level of effectiveness in explaining the inclination to embrace Internet of Things (IoT) in a DevOps setting. The R2 value of the endogenous construct serves as an indication, which is 0.504, beyond the threshold of 0.5. Nevertheless, the R2 statistic alone elucidates the extent to which the model can account for the variation in the data used for training (Saari et. al, 2021). To evaluate the predictive accuracy of our model for IoT adoption beyond the data it was trained on, we calculated Q2 values for key variables using a blindfolding strategy. The findings can be found in Table 4.2.6.

Table 4.2. 6 Relevance of the Model Prediction

Construct	Q^2 Predict
PT	0.126
IU	0.497

Source: *Raw, original information collected directly from the source*

Note: *PLS-SEM analysis is conducted utilizing the SMART PLS program, IU = "Intention to Use", PT = "Perceived Threat".*

Table 4.2.5 clearly indicates that the Q2 predicted values are greater than zero. Q2predict is utilized to validate that the predictions have surpassed the least sophisticated benchmark, referred to as "the average values obtained from the analysis sample" (Hair et. al, 2019). This highlights the model's ability to accurately anticipate result beyond the dataset it was trained on.

4.2.5 Analysis of Importance-Performance Maps (IMPA)

An importance-performance map analysis (IMPA) was undertaken to assess the impact and effectiveness of the constructs in relation to the endogenous construct.. The results of this analysis are shown in Figure 4.2.2 and Table 4.2.7. The external construct for which the total impacts significantly contribute to the explanation of the endogenous construct's variation is highlighted by the IMPA results (Saari et al., 2021).

Important information regarding the Internet of Things' (IoT) adoption in the DevOps environment may be found in the Importance Performance Map Analysis. With a positive effect size of 0.237 and a performance grade of 48.028, the research indicates that the degree

of convenience in utilizing IOT, also known as effort expectancy, has a considerable and extremely beneficial impact on its adoption.

However, facilitating conditions appear to have a minimal beneficial influence, indicated by an effect size of 0.03 and a performance score below the mean at 45.909. This suggests that they do not contribute significantly in determining adoption. Notably, the concepts of perceived harshness and vulnerability have a detrimental impact on the intention to utilize IoT, together with relatively low performance ratings.

Furthermore, the perceived threat has a substantial negative influence on adoption, with an effect size of -0.997. Nevertheless, it achieves a high score of 51.006 in terms of performance, indicating that it is a significant feature in the model. However, it also works as a big obstacle to adoption, as persons who view IoT as a threat are far less inclined to use it. In contrast, the impact of performance expectancy on adoption appears to be negligible, with a low effect size of 0.073 and a performance score of 44.63. This suggests that the expectations regarding the performance of IoT devices do not strongly influence their usage.

According to the statistics, social influence has a performance score of 46.139 and a moderate positive impact of 0.115, indicating that peer influence and social considerations are important factors in the decision to use IoT. With a performance score of 49.860, the 'Intention to Use' construct act as a standard for evaluating the effectiveness of the other model elements. While the mean performance score is 45.6, the mean absolute effect size is 0.3. In the model, constructs that achieve this performance level are considered to be performing above average.

In summary, the data suggests that the primary barrier to IoT adoption is the sense of threat, but effort expectancy and social influence play a significant role in its uptake. The performance scores offer a quantifiable assessment of the effectiveness of each construct in explaining the adoption. Surprisingly, perceived danger has the highest score, despite its adverse impact.

Constructs that have a performance score lower than the norm may require reassessment to determine their significance or may not be as crucial in comprehending adoption behavior.

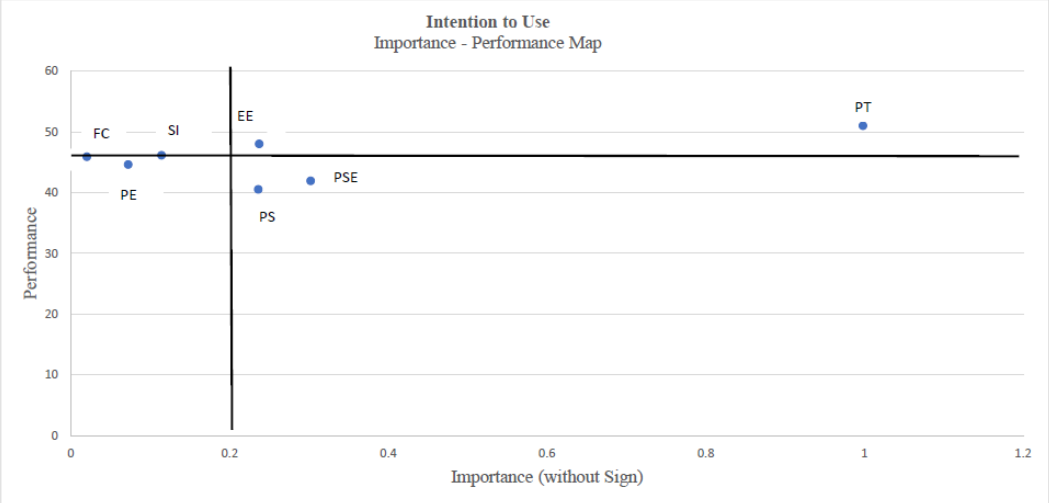
Table 4.2. 7 Analysis of Importance-Performance Map

	Unstandardized Total Effect (With Sign)	Unstandardized Total Effect (Without Sign)	Performance	LV Performance
EE	0.238	0.238	48.028	-
FC	0.03	0.03	45.909	-
PSE	-0.301	0.301	41.939	-
PS	-0.235	0.235	40.547	-
PT	-0.997	0.997	51.006	-
PE	0.073	0.073	44.63	-
SI	0.115	0.115	46.139	-
IU	-	-	-	49.860
Avg	-	0.3	45.6	

Source: *Raw, original information collected directly from the source*

Note: PLS-SEM analysis is conducted utilizing the SMART PLS program, *EE = "Effort Expectancy", FC = "Facilitating Conditions", IU = "Intention to Use", PSE = " Perceived Severity", PS = "Perceived Susceptibility", PT = "Perceived Threat", PE = "Performance Expectancy", SI = "Social Influence", Avg = "Average".*

Figure 4.2. 2 Analysis of Importance-Performance Map



Note: *PE = "Performance Expectancy", EE = "Effort Expectancy", SI = "Social Influence", FC = "Facilitating Conditions", PS = "Perceived Susceptibility", PT = "Perceived Threat", PSE = "Perceived Severity".*

Chapter 5 : DISCUSSION

Based on the Importance-Performance Map Analysis (IPMA) provided, the major factor affecting the adoption of IoT in DevOps appears to be the Perceived Threat associated with IoT systems. The investigated factors indicate that this particular component has the most pronounced negative impact (-0.997), which serves as a considerable obstacle to adoption. Furthermore, its performance score stands at the highest level of 51.006, which further emphasizes its crucial significance in influencing adoption behavior. **(Paprzycki et al., 2021)** The Next Generation IoT installations may face risks and vulnerabilities if security and privacy best practices are not carefully considered throughout the continuous delivery of services employing DevOps methodology.

Further, as discussed in the literature review, various threats, including privacy, reliability, and regulatory compliance, fundamentally affect the adoption of IoT in a DevOps-enabled environment. **(Thompson, 2019)** in his study sates that security, trust, and privacy are major challenges for IoT systems, and they should be resolved correctly. These risks are ignored by many developers because the cost and time are among other factors. IoT systems mostly operates in real time or changing environment and should have the ability to change and adapt. The study revealed that the support to make the smart IoT systems trustworthy and secure with DevOps practice is poor. Therefore, they developed a technique and an instrument that would secure the reliable operation of the IoT systems.

(Dorobantu & Halunga, 2020)

Have also revealed the vulnerability of the systems and have listed a few typical attacks including DDoS attacks, Botnets and of malware, data violation and opportunities that come from the Bluetooth links.

(Rajapakse et al., 2022)

Vulnerabilities in Continuous Integration (CI) systems pose a significant challenge within the DevOps pipeline. CI tools, essential for enabling this practice, are reported to be more vulnerable to security attacks compared to other tools. This vulnerability originates from tenants running their own code in the CI environment, resulting in an increased number of potential attack vectors. Furthermore, any security restrictions or vulnerabilities inside the Continuous Deployment (CD) pipeline itself pose significant security risks. The conventional design of CD pipelines frequently neglects the importance of security needs, and the participation of diverse tool suites and users throughout successive phases amplifies the susceptibility to malicious assaults. CD pipelines that have been compromised or misconfigured can lead to the introduction of malicious or experimental code into the production environment, which can pose significant hazards. Furthermore, in the context of DevSecOps, where collaboration among team members is crucial, granting equal access levels to Dev, Sec, and Ops members can introduce security challenges. Studies highlight that team members with access to CD pipeline configurations may inadvertently pose risks due to their lack of security knowledge and awareness, potentially leading to intentional or unintentional damages. Additionally, vulnerabilities such as

unencrypted connections and insecure environments further exacerbate security risks within the CD pipeline.

Therefore, eliminating threat to the implementation of IoT in DevOps, such as weak security of IoT systems, unsecure communication, security bug etc. has become necessary to adopt IoT in DevOps enabled environment.

(Ferry et al., 2019)

Suggest implementing autonomous cybersecurity monitoring as a method to implement security measures in the DevOps environment of IoT systems. The concept of Fast and Continuous Feedback (F&CF) is introduced to facilitate the prompt identification and resolution of difficulties in the development process, preventing any negative impact on the client. This F&CF activity enables teams to build monitoring and alerting services that are custom fitted. This is done to provide quick feedback from operations to development which improves early identification of security issues and bugs and prevents cyber-attacks at the time of deployment.

In addition, the thorough testing of the system's performance is very crucial. This process is aimed at identifying any weak points or flaws in security ensuring that the system is robust and secure. When this is done before any problems occur, it ensures that sensitive data is protected and that the IoT system is more reliable. **(Sand, 2016)** Security and performance testing are highly emphasized by the author supporting multichannel as well as behavior-driven testing approaches. They site importance of platform migration testing

and stress necessity of proper integration to testing automation frameworks and end-user performance analysis strategies. In a connected world, testing needs to be performed on applications running on multiple type of operating systems, devices, and in different locations, which best can be done in the cloud rather than on-premises. The author emphasizes the need to have access to test environments with high quality through testing partners. As competition grows swiftly companies have to keep current with trends and implement proper testing practices to improve their methodologies.

In addition to the perceived threat, the above analysis show that perceived susceptibility and perceived severity are critical factors that contribute to negative attitudes toward the adoption of IOT systems in DevOps environment. Perceived susceptibility, with a total effect of -0.235, indicates that as concerns about vulnerability increase, intention to use decreases by 0.235 units. Similarly, perceived severity, with a total effect of -0.301, suggests that for every unit increase in the seriousness of potential issues, intention to use decreases by 0.301 units. These results reflect the significance of resolving these issues in order to promote the increase in the usage intent and the diffusion of Internet of Things (IoT) ecosystem. This can provide the necessary guidance for designing the interventions that would address the perceived threats and that would, consequently, improve the intake of IoT technologies in DevOps contexts.

(Yadav et al., 2018)

Need of security in the field of IoT, particularly since the quantity of interconnected devices continues to rapidly increase. With millions transitioning to tens of billions, the risk of

exploiting vulnerabilities escalates, particularly in devices with inadequate designs or compromised data streams, posing threats to people's health and safety. Additionally, IoT setups often comprise clusters of similar devices, amplifying the impact of any security flaw across numerous devices with identical features. Furthermore, privacy is a paramount consideration alongside security. Beyond authenticity, trustworthiness, and confidentiality, ensuring discriminatory access, restricting data sharing, and securing business communications involving smart objects are vital requirements. These aspects safeguard sensitive information and prevent unauthorized access or misuse.

Another factor influencing the adoption of IoT in DevOps is Effort Expectancy. Despite its positive total effect (0.238), this factor is associated with a relatively high-performance score (48.028), indicative of its substantial impact on adoption behavior within the studied environment. Effort Expectancy is essentially the term used to describe how easy people think it is to use IoT technologies within a DevOps environment. If the IoT system is complex, it would be difficult to use it in a DevOps-enabled environment, as the effort needed would be more.

(Colliander, 2022)

DevSecOps encounters challenges in resource-limited scenarios, like the Internet of Things (IoT). The decentralized and diverse character of IoT systems gives rise to substantial security apprehensions. These are complex to manage because the network is volatile. The different types of IoT devices, from a variety of vendors, connecting through diverse communication channels, sending different processed data from different environments

creates a complex varying system **(Thompson, 2019).** The testing industry must respond to the challenges of the emerging Internet of Things (IoT) sector, which is intricate and new. **(Gomez & Bajaj, 2019)** mentioned IoT is known for being not standardized; this is its heterogeneity. It also has security issues and privacy concerns as well as interoperability complexity among other things such as test environment architecture. All these factors mean that there are many different combinations or versions of tests needed – this makes them expensive and hard to define because they have unknowns in them too. The need to protect several technologies, such as physical devices, wireless transmission, mobile, cloud architectures, and their connection with other technologies, has made IoT device security more complex. **(Al-Garadi et al., 2020).**

In the Importance-Performance Map, when "Social Influence" is analyzed, it shows a strong positive effect (0.115) on adopting IoT in DevOps. This means that higher social influence can be associated with greater chances of adoption hence underscoring its crucial part in fostering adoption behavior. Its significance during adoption process is evident at 46.139 performance score but what should be realized is that lessening the desire for using IoT technology may occur if there happens to be decline or decrease in social influences among people who do not work together closely often which emphasizes social dynamics in technology adoption within DevOps environments.

(López-Peña et al., 2020)

In contrast, individuals who are unaccustomed to collaborating must now work together, requiring the definition of new processes, activities, and their automation. By tackling these obstacles, we empower engineers to actively monitor the systems they build, allowing for prompt and continuous feedback. (**Claps, Svensson, & Aurum, 2015)** outline the challenges encountered by organizations that implemented the Continuous Deployment process from the perspectives of technology and social issues. The two scholars support the claim that a software development methodology would be inefficient if it is technologically acceptable but socially inappropriate. Similarly, an SDM would not be used extensively, even if it is socially acceptable, if it has issues with technicality. To guarantee the successful implementation of the CD, it is critical to address both technical and social hurdles.

According to the **(Maroukian & Gulliver, 2020a),** a roadmap to DevOps practices and principles must be developed, with a clear transition between stages. The latter should include the team structure and convergence forms and leadership that support agency and lean in those forms. Moreover, while a DevOps lead...a true continuous influencer, and not tied to individual projects, is mandated for traction, there are some challenges that the company must consider. These are poor and exhausting communication, insufficient bi-directional flow of information, artifacts, and knowledge, cemented company culture, and mere overdrawing of operations in requirements decomposition and specification.

From Table 4.2.5 it is evident that the Q2 predict values are higher than zero. This demonstrates that the forecasts outperform the most basic estimate, which is determined by taking the average of the data. This proves that the model's predictions are relevant even when applied to new data.

Chapter 6 : CONCLUSION

It is widely recognized that DevOps has increased the speed and reliability of delivery in applications. Numerous stories of success in improving customer service have been pegged on the idea of DevOps. However, DevOps is still not widely adopted in the IoT field, where many components such as hardware, software and firmware are involved. As a result, people do not fully utilize DevOps methods for IoT applications. This means missed function capacity potential and efficiency driving new ideas for business growth. Therefore, we must apply DevOps principles and culture to IoT deployment at once. This will ensure that productivity grows, and time-to-market is reduced for IoT products and applications. Facilitating the connection between development and operation in IoT projects can bring greater efficiency and development potential.

In order to investigate the factors impacting the adoption of IoT systems in DevOps contexts, a theoretical framework was developed by combining the Technology Threat Avoidance Model (TTAT) with the Unified Theory of Acceptance and Use of Technology (UTAUT). Variables used in UTAUT characteristics like effort expectancy, performance expectancy, social influence, and facilitating condition. Most of these variables have been considered when preparing the questionnaire and TTAT variables like perceived susceptibility, perceived severity, and perceived threat.

Afterward, G* Power software was used for determining the ideal sample size for the research study model. G* Power recommends 370 samples. However, the sample size has

been chosen to accomplish accurate statistical outcomes. Therefore, 450 people engaged in the research. Due to the specialized target group who were specialized in using such modern technologies, the study used purposive sampling. Further, data was gathered via a designed questionnaire. SMART PLS software and the PLS-SEM method were used for analysis to assess the research results.

The study revealed that the adoption of IoT systems in DevOps-enabled environments is driven inversely by perceived threat and directly by effort expectancy. The total effect of perceived threat is negative, implying that security and risk concerns related to these systems are of paramount importance. The total effect of effort expectancy, meanwhile, is positive, implying that perceived effort should be minimized when it comes to using IoT systems. Furthermore, perceived severity and perceived susceptibility have a direct effect on intention to use the system — supported by perceived threat as a mediator. While other factors, such as facilitating condition, performance expectancy, and social influence, also influence the adoption of IoT systems, they are of a more limited nature.

It was reported from the Important-Performance Map Analysis that perceived threat is the most important indicator for hindering adoption of IoT in DevOps enabled environment. When the perceived threat falls by 1 unit, for as from 51 to 50, the desire to adopt IoT in a DevOps environment increases by 0.997 units, from 50 to 50.997. This represents the most significant improvement in the performance of our intended structure.

According to the IPMA results, it is advised to address the perceived threat in order to enhance the adoption of IoT in a DevOps context. To address DevSecOps challenges in challenging situations, it is essential to develop specialized frameworks and solutions specifically designed to resolve these issues. Two potential solutions are the self-service cybersecurity monitoring service for IoT systems created by **(Díaz et al., 2019)** and the security architecture for IoT devices with double authentication offered by (Sridhar & Smys, 2017). Nevertheless, the majority of organizations fail to adequately tackle challenges such as resolving complex infrastructures or navigating limited contexts. The research on applying DevSecOps in such infrastructures is currently in the scientific phase, and a complete solution has not yet been developed **(Colliander, 2022).**

Effort Expectancy is another significant variable in the adoption IoT in DevOps. According to Yun, He and Jun effort expectancy is concerned with people's opinions about how simple it is to use and accept a technological system. This includes the complexity of setup, how the tool integrates other technologies and processes and general ease of use, among others. Individuals are more likely to adopt systems they find easy to use for by reducing the perceived effort in adopting IoT systems makes them more adoptable. **(López-Peña et al., 2020)** Continuous monitoring is a big problem in the era of IoT. Systems Researchers have carried out comprehensive reviews of the literature and many case studies to understand the main obstacles to adopting DevOps in embedded systems, as well the widespread distribution of these ideas in the web location. These obstacles include the technical complexities inherent in such a distributed nature of IoT systems as across cloud and edge sides — (and likewise for their people); the necessity and unfamiliarity of working together

that everyone must collaborate, a very different process and practice can bring about change. By addressing these issues, developers can effectively oversee the systems they create through on-demand monitoring and self-service mechanisms, resulting in prompt and constant feedback. Although there have been advancements in automating the setup and distribution of IoT devices, the task of continuous monitoring still requires significant effort.

The result state that social influence is a also a important variable affecting IoT adoption in DevOps. Big IT companies can exert social influence by publishing success stories for IoT projects using IoT and DevOps. Events and conference can socially influence the next generation organization it follows that IoT in DevOps will become more and more popular. **(Pereira, de Senna Carneiro, & Figueiredo, 2021)** state that many people trace the origins of DevOps back to a meeting between Patrick Debois and Andrew Shaffer in Toronto in 2008, which is often recognized as ground zero for this movement. Another major event was the presentation titled "10 Deploys a day" by John Allspaw and Paul Hammond on Flickr. This was the inspiration behind Debois organizing DevOpsDays.

(Wiedemann et al., 2019) Central to the success of DevOps teams is their ability to share knowledge and information. Sharing in this context refers to the sharing of successes and fails with teams internally, to organizations as a whole and to the entire industry. Unlike other methodologies, this habit of sharing has grown to be a culture within the DevOps setting. DevOps teams swap roles, allowing developers to handle operations and vice versa, fostering a deeper understanding of each other's responsibilities. Additionally, industries

share their experience at conferences and local community events that are becoming popular across the globe, like DevOpsDays.

The future studies should, however, go deeper to understand the security aspects of IoT, this includes creation of robust security frameworks, security awareness training evaluation, putting regulations in place, and improving the risk management strategy that would be ideal for IoT deployment in DevOps workflows. By performing in-depth investigations in these areas, researchers will be able to contribute significantly in enhancing the security of Internet of Things (IoT) systems and facilitating their successful adoption within DevOps environments.

BIBLIOGRAPHY

Abdalla, P.A. and Varol, C., 2020, June. Testing IoT security: The case study of an ip camera. *In 2020 8th International Symposium on Digital Forensics and Security* (ISDFS) (pp. 1-5). IEEE.doi: 10.1109/ISDFS49300.2020.9116392

Abidi, N., Yanamandra, R., Nair, H.K., Al Nasar, M.R., & Khassawneh, O. (2023, March). Impact of IoT and Resource-Based View on Digital Business: The Role of Strategic Thinking Leadership. *In 2023 International Conference on Business Analytics for Technology and Security (ICBATS) (pp. 1-7). IEEE*. doi: 10.1109/ICBATS57792.2023.10111305

Akhilesh, R., Bills, O., Chilamkurti, N. and Chowdhury, M.J.M., 2022. *Automated penetration testing framework for smart-home-based IoT devices*. Future Internet, 14(10), p.276. doi: doi.org/10.3390/fi14100276

Aktaş, G., Konukoğlu, E., & Aydın, Y. (2023, November). Monitoring Approach and Framework Development for System Tracking in IoT Devices. *In 2023 14th International Conference on Electrical and Electronics Engineering (ELECO) (pp. 1-4). IEEE*. doi: 10.1109/ELECO60389.2023.10416017

Alhaidari, F., Rahman, A., & Zagrouba, R. (2023). Cloud of Things: Architecture, applications and challenges. *Journal of Ambient Intelligence and Humanized Computing, 14(5), 5957-5975.* doi:10.1007/s12652-020-02448-3

Alladi, Tejasvi, Vinay Chamola, Biplab Sikdar, & Kim-Kwang Raymond Choo. (2020). Consumer IoT: Security vulnerability case studies and solutions. *IEEE Consumer Electronics Magazine*, 9(2), 17-25.doi: 10.1109/MCE.2019.2953740

Aman, A. H. M., Yadegaridehkordi, E., Attarbashi, Z. S., Hassan, R., & Park, Y. J. (2020). A survey on trend and classification of internet of things reviews. *IEEE Access*, 8, 111763-111782.doi: 10.1109/ACCESS.2020.3002932

Asad, M., Moustafa, A., & Yu, C. (2020). A critical evaluation of privacy and security threats in federated learning. Sensors, 20(24), p.7182. doi:10.3390/s20247182.

Bacudio, A.G., Yuan, X., Chu, B.T.B., & Jones, M. (2011). An overview of penetration testing. *International Journal of Network Security & Its Applications*, 3(6), p.19. Retrieved from link (Accessed: April 13, 2023).

Battina, D.S. (2017). Best practices for ensuring security in DevOps: A case study approach. *International Journal of Innovations in Engineering Research and Technology*, 4(11), 38-45. Available at: https://www.researchgate.net/publication/357033114_BEST_PRACTICES_FOR_ENSURING_SECURITY_IN_DEVOPS_A_CASE_STUDY_APPROACH (Accessed: April 13, 2024)

Bijwe, A., & Shankar, P. (2023). DevOps culture and practices for IoT applications. Retrieved from https://www.biogecko.co.nz/admin/uploads/NC-SCO-117.pdf (Accessed: February 21, 2024).

Bîrlog, I.A., Borcan, D.M., & Covrig, G.M. (2020). Internet of things hardware and software. Informatica Economica, 24(2), 54-62. doi:10.24818/issn14531305/24.2.2020.05

Brous, P., Janssen, M., & Herder, P. (2020). The dual effects of the Internet of Things (IoT): A systematic review of the benefits and risks of IoT adoption by organizations. *International Journal of Information Management*, 51, p.101952. doi: 10.1016/j.ijinfomgt.2019.05.008

Chandan, A. R., & Khairnar, V. D. (2018, July). Security testing methodology of IoT. In 2018 *International Conference on Inventive Research in Computing Applications* (ICIRCA) (pp. 1431-1435). IEEE. doi: 10.1109/ICIRCA.2018.8597192

Cheruvu, S., Kumar, A., Smith, N., & Wheeler, D.M. (2020). Demystifying Internet of Things Security: Successful IoT Device/Edge and Platform Security Deployment (p. 488). Springer Nature. doi:10.1007/978-1-4842-2896-8.

Chougul, S. (2019). IoT Device Penetration Testing. Retrieved from https://owasp.org/www-chapter-pune/meetups/2019/August/IoT_Device_Pentest_by_Shubham_Chougule.pdf (Accessed: April 13, 2024).

Chowdhary, A., Huang, D., Mahendran, J.S., Romo, D., Deng, Y. and Sabur, A., 2020, December. Autonomous security analysis and penetration testing. *In 2020 16th International Conference on Mobility, Sensing and Networking* (MSN) (pp. 508-515). IEEE.doi: 10.1109/MSN50589.2020.00086

Chu, G., & Lisitsa, A. (2018, June). Penetration testing for Internet of Things and its automation. *In 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)* (pp. 1479-1484). IEEE. doi:10.1109/HPCC/SmartCity/DSS.2018.00244

Claps, G. G., Svensson, R. B., & Aurum, A. (2015). On the journey to continuous deployment: Technical and social challenges along the way. *Information and Software Technology*, 57, 21-31. doi: doi.org/10.1016/j.infsof.2014.07.009

Colliander, C. (2022). Challenges of DevSecOps (Master's thesis). *University of Helsinki, Faculty of Science*. Retrieved from http://hdl.handle.net/10138/342887. (Accessed: March 11, 2024).

Cowart, N. (2019). *What is the Connection Between IoT and DevOps?* Retrieved from https://www.kovair.com/blog/connection-between-IoT-and-devops(Accessed: February 12, 2024).

Critchley, A. and Latonick, H. (2020) 'False Alarms and Close Calls: The Analysis and Verification of Ripple20 and its Ripple Effect', Foresite. Available at: https://finitestate.io/ripple-20-verification (Accessed: 13 April 2024).

Cybersecurity Monitoring as Enabler for DevSecOps. *IEEE Access, 7, 100283*-

Damanpour, F., & Gopalakrishnan, S. (1998). Theories of organizational structure and innovation adoption: The role of environmental change. *Journal of Engineering and Technology Management*, 15(1), 1–24. doi: 10.1016/S0923-4748(97)00029-5.

Davis, B.D., Mason, J.C. and Anwar, M., 2020. Vulnerability studies and security postures of IoT devices: A smart home case study. *IEEE Internet of Things Journal*, *7*(10), pp.10102-10110. doi: 10.1109/JIOT.2020.2983983

de França, B.B.N., Jeronimo, H. and Travassos, G.H., 2016, September. Characterizing DevOps by hearing multiple voices. I*n Proceedings of the XXX Brazilian Symposium on Software Engineering* (pp. 53-62). Doi: doi.org/10.1145/2973839.2973845

Debois P (2008) Agile infrastructure and Operations: How infra-gile are you? *In: Agile 2008 Conference, pp 202–207*. doi: 10.1109/Agile.2008.42

Denis, M., Zena, C. and Hayajneh, T., 2016, April. Penetration testing: Concepts, attack methods, and defense strategies. *In 2016 IEEE Long Island Systems, Applications and Technology Conference (LISAT) (pp. 1-6)*. IEEE.doi: 10.1109/LISAT.2016.7494156

Díaz, J., López-Fernández, D., Pérez, J., & González-Prieto, Á. (2021). Why are many businesses instilling a DevOps culture into their organization? *Empirical Software Engineering*, 26, 1-50. doi:10.1007/s10664-020-09919-3

Diaz, J., Pérez, J. E., Lopez-Peña, M. A., Mena, G. A., & Yagüe, A. (2019). Self-service cybersecurity monitoring as enabler for DevSecOps. *IEEE Access, 7, 100283-100295*. doi:10.1109/ACCESS.2019.2930000

Dofe, J., Frey, J., & Yu, Q. (2016, May). Hardware security assurance in emerging IoT applications. *In 2016 IEEE International Symposium on Circuits and Systems (ISCAS) (pp. 2050-2053). IEEE*. doi: 10.1109/ISCAS.2016.7538981

Dorobantu, O. G., & Halunga, S. (2020). Security threats in IoT. *In 2020 International Symposium on Electronics and Telecommunications (ISETC) (pp. 1-4). IEEE*. doi: 10.1109/ISETC50328.2020.9301127

Elena and Anna. 2022. Internet of things (IOT) security: Challenges and best practices. Apriorit. Available at: https://www.apriorit.com/white-papers/513-IoT-security (Accessed: February 26, 2023).

Engebretson, P. (2013). The basics of hacking and penetration testing: ethical hacking and penetration testing made easy. Elsevier. Retrieved from https://www.google.co.in/books/edition/The_Basics_of_Hacking_and_Penetration_Te/69dEUBJKMiYC?hl=en&gbpv=1 (Accessed: February 12, 2024).

Ferry, N. and Nguyen, P.H., 2019, September. Towards model-based continuous deployment of secure IoT systems. *In 2019 ACM/IEEE 22nd International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)* (pp. 613-618). IEEE.doi: 10.1109/MODELS-C.2019.00093

Ferry, N., Solberg, A., Song, H., Lavirotte, S., Tigli, J.Y., Winter, T., Muntés-Mulero, V., Metzger, A., Rios Velasco, E., & Castelruiz Aguirre, A. (2019). Enact: Development, operation, and quality assurance of trustworthy smart IoT systems. In Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment: *First International Workshop, DEVOPS 2018, Chateau de Villebrumier, France, March 5-6, 2018, Revised Selected Papers 1 (pp. 112-127). Springer International Publishing*. Retrieved from https://link.springer.com/chapter/10.1007/978-3-030-06019-0_9 (Accessed: February 12, 2024).

Garg, H. and Dave, M., 2019, April. Securing IoT devices and securely connecting the dots using rest api and middleware. In 2019 *4th International Conference on Internet of Things: Smart Innovation and Usages* (IoT-SIU) (pp. 1-6). IEEE. doi: 10.1109/IoT-SIU.2019.8777334

Ghantous, G. B., & Gill, A. (2020, August). The DevOps reference architecture Evaluation: A design Science research case study. *In 2020 IEEE International Conference on Smart Internet of Things (SmartIoT) (pp. 295-299). IEEE*. doi:10.1109/SmartIoT49966.2020.00052

Gomez, A. K., & Bajaj, S. (2019, October). Challenges of testing complex Internet of Things (IoT) devices and systems. *In 2019 11th international conference on knowledge and systems engineering (KSE) (pp. 1-4). IEEE*. doi: 10.1109/KSE.2019.8919324

Griffin, R. P., Tatar, U., & Yankson, B. (Eds.). (2022). *Proceedings of the 17th International Conference on Cyber Warfare and Security*. State University of New York at Albany, Albany, New York, USA, 17-18 March 2022. *A conference managed by ACI, UK*. Retrieved from https://papers.academic-conferences.org/index.php/iccws/issue/view/2/1(Accessed: February 17, 2024).

Gupta, B. B., & Quamara, M. (2020). An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols. *Concurrency and Computation: Practice and Experience, 32(21), e4946*. doi:10.1002/cpe.4946

Guşeilă, L. G., Bratu, D. V., & Moraru, S. A. (2019, August). Continuous testing in the development of IoT applications. *In 2019 International Conference on Sensing and Instrumentation in IoT Era (ISSI) (pp. 1-6). IEEE*. doi:10.1109/ISSI47111.2019.9043692

Hair, J.F., Risher, J.J., Sarstedt, M. and Ringle, C.M., 2019. When to use and how to report the results of PLS-SEM. *European business review*, 31(1), pp.2-24. doi:doi.org/10.1108/EBR-11-2018-0203

Hiremath, O. 2023, From Penetration Testing to AppSec/DevSecOps: A Guide to Staying Ahead of the Curve, viewed 13 April 2024, https://www.guardrails.io/blog/from-penetration-testing-to-appsec-devsecops-a-guide-to-staying-ahead-of-the-curve.

Iqbal, W., Abbas, H., Daneshmand, M., Rauf, B., & Bangash, Y. A. 2020, An in-depth analysis of IoT security requirements, challenges, and their countermeasures via software-

defined security. *IEEE Internet of Things Journal*, 7(10), pp. 10250-10276. doi:10.1109/JIOT.2020.2997651

Jevtic, G. (2019, May 9). *17 Best Security Penetration Testing Tools The Pros Use.* Retrieved from https://phoenixnap.com/blog/best-penetration-testing-tools

Jha, A. V., Teri, R., Verma, S., Tarafder, S., Bhowmik, W., Kumar Mishra, S., Appasani, B., Srinivasulu, A., & Philibert, N. (2023). From theory to practice: Understanding DevOps culture and mindset. *Cogent Engineering*, 10(1), 2251758. doi:10.1080/23311916.2023.2251758

Johari, R., Kaur, I., Tripathi, R. and Gupta, K., 2020, October. Penetration testing in IoT network. In 2020 *5th International Conference on Computing, Communication and Security (ICCCS)* (pp. 1-7). IEEE.doi: 10.1109/ICCCS49678.2020.9276853

Johari, R., Kaur, I., Tripathi, R. and Gupta, K., 2020, October. Penetration testing in IoT network. *In 2020 5th International Conference on Computing, Communication and Security (ICCCS)* (pp. 1-7). IEEE.doi: 10.1109/ICCCS49678.2020.9276853

Jokela, A. (2019). Agile and Lean processes on an IoT development platform: A Case Study. Retrieved from https://aaltodoc.aalto.fi/server/api/core/bitstreams/0004465c-5af0-42b9-9a93-902056ff842a/content (Accessed: February 12, 2024).

Karamitsos, I., Albarhami, S. and Apostolopoulos, C., 2020. Applying DevOps practices of continuous automation for machine learning. Information, 11(7), p.363.doi: doi.org/10.3390/info11070363

Karapantelakis, A., Liang, H., Wang, K., Vandikas, K., Inam, R., Fersman, E., Mulas-Viela, I., Seyvet, N., & Giannokostas, V. (2016, August). DevOps for IoT applications using cellular networks and cloud. *In 2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud) (pp. 340-347). IEEE*. doi:10.1109/FiCloud.2016.55

Khan, M. S., Khan, A. W., Khan, F., Khan, M. A., & Whangbo, T. K. (2022). Critical challenges to adopt DevOps culture in software organizations: *A systematic review. IEEE Access, 10, 14339-14349.* doi:10.1109/ACCESS.2022.3145970

Khan, M. Z., Alhazmi, O. H., Javed, M. A., Ghandorh, H., & Aloufi, K. S. (2021). Reliable Internet of Things: Challenges and future trends. *Electronics,* 10(19), 2377. doi:10.3390/electronics10192377

Khan, R., Khan, S.U., Zaheer, R., & Khan, S. (2012). Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges. *In 2012 10th International Conference on Frontiers of Information Technology (pp. 257-260). IEEE*. doi:10.1109/FIT.2012.53

Kokila, M., & Reddy, S. (2024). Authentication, access control, and scalability models in Internet of Things security: A review. Cyber Security and Applications, p.100057. doi: 10.1016/j.csa.2024.100057

Koohang, A., Sargent, C. S., Nord, J. H., & Paliszkiewicz, J. (2022) 'Internet of Things (IoT): From awareness to continued use', *International Journal of Information Management*, 62, p.102442. doi:10.1016/j.ijinfomgt.2021.102442.

Kumar, M., Kumar, A., Verma, S., Bhattacharya, P., Ghimire, D., Kim, S. H., & Hosen, A. S. (2023). Healthcare Internet of Things (H-IoT): Current trends, future prospects, applications, challenges, and security issues. *Electronics*, 12(9), 2050. doi:10.3390/electronics12092050

Kumar, S., Tiwari, P., & Zymbler, M. (2019). Internet of Things is a revolutionary approach for future technology enhancement: *A review. Journal of Big Data,* 6(1), 1-21. doi:10.1186/s40537-019-0268-2

Lampropoulos, G., Siakas, K., & Anastasiadis, T. (2019). Internet of things in the context of industry 4.0: An overview. **International Journal of Entrepreneurial Knowledge*, *pp. 4-19.

Lekić, M., & Gardašević, G. (2018). IoT sensor integration to Node-RED platform. *In 2018 17th International Symposium Infoteh-Jahorina (Infoteh) (pp. 1-5). IEEE*. doi:10.1109/INFOTEH.2018.8345544

Liang, H., & Xue, Y. (2009). Avoidance of information technology threats: a theoretical perspective. MIS Quarterly, 33(1), 71-90.doi: doi.org/10.2307/20650279

Ling, Z., Liu, K., Xu, Y., Gao, C., Jin, Y., Zou, C., Fu, X. and Zhao, W., 2018. IoT security: An end-to-end view and case study. *arXiv preprint arXiv:1805.05853*.doi:doi.org/10.48550/arXiv.1805.05853

López-Peña, M.A., Díaz, J., Pérez, J.E., & Humanes, H. (2020). DevOps for IoT systems: Fast and continuous monitoring feedback of system availability. *IEEE Internet of Things Journal*, 7(10), 10695-10707. doi:10.1109/JIOT.2020.3012763.

Maier, A., Sharp, A., & Vagapov, Y. (2017, September). Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things. *In 2017 Internet Technologies and Applications (ITA) (pp. 143-148). IEEE*. doi:10.1109/ITECHA.2017.8101926

Maroukian, K., & Gulliver, S. (2020a). Exploring the link between leadership and DevOps practice and principle adoption. *Advanced Computing: An International Journal, 11(4)*. doi:10.5121/acij.2020.11401

Maroukian, K., & Gulliver, S. R. (2020b, November). The link between transformational and servant leadership in DevOps-oriented organizations. *In Proceedings of the 2020 European Symposium on Software Engineering (pp. 21-29)*. doi:10.1145/3393822.3432340

Mohd Aman, A. H., Shaari, N., & Ibrahim, R. (2021). Internet of things energy system: Smart applications, technology advancement, and open issues. *International Journal of Energy Research, 45(6), 8389-8419.* doi:10.1002/er.6451

Mrabet, H., Belguith, S., Alhomoud, A., & Jemai, A. (2020). A survey of IoT security based on a layered architecture of sensing and data analysis. *Sensors, 20(13), 3625*. doi:10.3390/s20133625

Owens, J. (2023). Practical IoT Cybersecurity: Hands-on Penetration Testing for Smart Connected Devices (pp. 133-134). Retrieved from https://www.amazon.com/Practical-IoT-Cybersecurity-Hands-Penetration/dp/B0CGG83NKV Accessed on April 13, 2024.

Pang, C., Hindle, A., & Barbosa, D. (2020, June). Understanding DevOps education with grounded theory. *In Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering Education and Training (pp. 107-118).* doi:10.1145/3377814.3381711

Paprzycki, M., Ganzha, M., Wasielewska, K., & Lewandowski, P. (2021). DevSecOps methodology for NG-IoT ecosystem development lifecycle–ASSIST-IoT perspective. *Journal of Computer Science and Cybernetics, 37(3), 321-337*. doi:10.15625/1813-9663/37/3/16245.

Pereira, I.M., Carneiro, T., & Figueiredo, E. (2021, March). A systematic review on the use of DevOps in Internet of Things software systems. *In Proceedings of the 36th Annual ACM Symposium on Applied Computing* (pp. 1569-1571). doi:10.1145/3412841.3442126.

Pereira, I.M., de Senna Carneiro, T.G., & Figueiredo, E. (2021, June). Understanding the context of IoT software systems in DevOps. *In 2021 IEEE/ACM 3rd International Workshop on Software Engineering Research and Practices for the IoT (SERP4IoT) (pp. 13-20). IEEE*. doi: 10.1109/SERP4IoT52556.2021.00009

Rajapakse, R. N., Zahedi, M., Babar, M. A., & Shen, H. (2022). Challenges and solutions when adopting DevSecOps: A systematic review. *Information and Software Technology*, 141, 106700. doi:10.1016/j.infsof.2021.106700

Rose, K., Eldridge, S. and Chapin, L., 2015. The internet of things: An overview. *The internet society (ISOC),* 80(15), pp.1-53.

Routh, K., & Pal, T. (2018, February). A survey on technological, business and societal aspects of Internet of Things by Q3, 2017. *In 2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU) (pp. 1-4). IEEE*. doi: 10.1109/IoT-SIU.2018.8519898

Saari, U.A., Damberg, S., Frömbling, L. and Ringle, C.M., 2021. Sustainable consumption behavior of Europeans: The influence of environmental knowledge and risk perception on environmental concern and behavioral intention. *Ecological Economics*, *189*, p.107155. doi: doi.org/10.1016/j.ecolecon.2021.107155

Sand, B. (2016). IoT testing-the big challenge why, what and how. In Internet of Things. IoT Infrastructures: Second International Summit, IoT 360° 2015, Rome, Italy, October

27-29, 2015, Revised Selected Papers, Part II (pp. 70-76). *Springer International Publishing*. doi:10.1007/978-3-319-47075-7_9

Selgert, F., 2020. Cynefin Framework, DevOps and Secure IoT: Understanding the Nature of IoT Systems and Exploring Where in the DevOps Cycle Easy Gains Can Be Made to Increase Their Security. In *Computer Safety, Reliability, and Security. SAFECOMP 2020 Workshops: DECSoS 2020, DepDevOps 2020, USDAI 2020, and WAISE 2020, Lisbon, Portugal, September 15, 2020, Proceedings 39* (pp. 255-265). *Springer International Publishing*. doi: https://doi.org/10.1007/978-3-030-55583-2_19

Sethi, P., & Sarangi, S.R. (2017). Internet of Things: Architectures, Protocols, and Applications. Journal of Electrical & Computer Engineering. doi:10.1155/2017/9324035

Shakdher, A., Agrawal, S. and Yang, B., 2019, May. Security vulnerabilities in consumer IoT applications. In 2019 *IEEE 5th Intl conference on big data security on cloud (BigDataSecurity), IEEE intl conference on high performance and smart computing,(HPSC) and IEEE intl conference on intelligent data and security (IDS*) (pp. 1-6). IEEE. doi: 10.1109/BigDataSecurity-HPSC-IDS.2019.00012

Sherman, J. (2022) 'Vulnerabilities: A Look Back at the Top 12 IoT Exploits of 2021 (Part 1)', Foresite. Available at: https://finitestate.io/blog/top-12-IoT-exploits-of-2021-p1 (Accessed: 15 April 2024).

Sridhar, S., & Smys, S. (2017). Intelligent security framework for IoT devices: Cryptography-based end-to-end security architecture. *In 2017 International Conference on Inventive Systems and Control (ICISC)*. doi: 10.1109/ICISC.2017.8068718

Tawalbeh, L. A., Muheidat, F., Tawalbeh, M., & Quwaider, M. (2020). IoT Privacy and security: Challenges and solutions. *Applied Sciences, 10(12), 4102*. doi:10.3390/app10124102

Thompson, A. R. (2019). Tool support for risk-driven planning of trustworthy smart IoT systems within DevOps (Master's thesis). Retrieved from https://www.duo.uio.no/bitstream/handle/10852/73279/master.pdf?sequence=1&isAllowed=y (Accessed: March 11, 2024).

Uy, N.Q., & Nam, V.H. (2019, December). A comparison of AMQP and MQTT protocols for Internet of Things. *In 2019 6th NAFOSTED Conference on Information and Computer Science (NICS) (pp. 292-297). IEEE*. doi:10.1109/NICS48868.2019.9023812

Vanwell, J. (2021) 'IoT Security Breaches: 4 Real-World Examples', Conosco. Available at: https://conosco.com/industry-insights/blog/IoT-security-breaches-4-real-world-examples (Accessed: 13 April 2024).

Veluvarthi, R., Rameswarapu, A., Kalyan, K.S., Piri, J., & Acharya, B. (2023, March). Security and Privacy Threats of IoT Devices: A Short Review. *In 2023 4th International Conference on Signal Processing and Communication* (ICSPC) (pp. 32-37). IEEE. doi:10.1109/ICSPC57692.2023.10125863.

Venkatesh, V., Morris, M. G., Davis, G. B., & Davis, F. D. (2003). User acceptance of information technology: Toward a unified view. MIS Quarterly, 27(3), 425-478.doi: https://doi.org/10.2307/30036540

Visoottiviseth, V., Akarasiriwong, P., Chaiyasart, S., & Chotivatunyu, S. (2017, November). PENTOS: Penetration testing tool for Internet of Thing devices. In TENCON 2017-2017 IEEE Region 10 Conference (pp. 2279-2284). IEEE. doi:10.1109/TENCON.2017.8228241

Wiedemann, A., Forsgren, N., Wiesche, M., Gewald, H., & Krcmar, H. (2019). The DevOps phenomenon: An executive crash course. *Queue, 17(2), 93-112*. doi: doi.org/10.1145/3329781.3338532

Xu, T., Wendt, J.B. and Potkonjak, M., 2014, November. Security of IoT systems: Design challenges and opportunities. In *2014 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)* (pp. 417-423). IEEE. doi: 10.1109/ICCAD.2014.7001385

Yadav, E. P., Mittal, E. A., & Yadav, H. (2018, February 23). IoT: Challenges and issues in Indian perspective. *In 2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU) (pp. 1-5). IEEE*. doi:10.1109/IoT-SIU.2018.8519869

Yadav, G., Allakany, A., Kumar, V., Paul, K., & Okamura, K. (2019, July). Penetration testing framework for IoT. I*n 2019 8th International Congress on Advanced Applied Informatics (IIAI-AAI)* (pp. 477-482). IEEE. doi: 10.1109/IIAI-AAI.2019.00104

Zhang, Z.K., Cho, M.C.Y., Wang, C.W., Hsu, C.W., Chen, C.K. and Shieh, S. 2014. IoT security: ongoing challenges and research opportunities. *In 2014 IEEE 7th International Conference on Service-Oriented Computing and Applications*, pp. 230-234. IEEE. doi: 10.1109/SOCA.2014.58

ANNEXURE – QUESTIONNAIRE

Demographics

1. Place
    a) Chennai
    b) Mumbai
    c) Delhi
    d) Kolkata
    e) Bangalore
2. Gender
    a) Male
    b) Female
3. Age
    a) 18-30 years
    b) 30-40 years
    c) 40-50 years
    d) Above 50 years
4. Firm Age
    a) Less than or equal to 25 years
    b) More than 25 years
5. Technology Adoption
    a) High
    b) Low
6. R&D Investment
    a) High
    b) Low

Please rate the Below Statements

(From 1 – strongly disagree to 7 – strongly agree)

Construct	Indicator	1	2	3	4	5	6	7
Performance Expectancy *(Venkatesh et al., 2012)*	PE01 - I expect that using IoT systems in our DevOps practices will significantly enhance our operational efficiency							
	PE02 - Integrating IoT systems into our workflow will improve our ability to monitor and manage operations in real time							
	PEO3 - The adoption of IoT systems will lead to better quality outputs in our projects							
Effort Expectancy *(Venkatesh et al., 2012)*	EE01 – I believe that integration IOT systems into our DevOps practices would be easy for us to manage							
	EE02 – I feel confident that I can effectively use IOT technologies with minimal effort							
	EE03 – Learning to operate IOT systems within our DevOps environment requires little effort from our team							

Facilitating Condition *(Venkatesh et al., 2012)*	FC01 - I have the necessary resources to use/implement IOT systems							
	FC02 – I/My team have the knowledge necessary to use/adopt IOT systems							
	FC03 – My company/business unit facilitates the use of IOT systems through various supporting initiatives							
Social Influence *(Venkatesh et al., 2012)*	SI01 - Peers who influence my behavior think that I should use IOT systems.							
	SI02 – My peers who use IOT systems have a more positive attitude towards the use of IOT systems in their job.							
	SI03 - People who are important to me think that I should use IOT systems.							
Perceived Susceptibility *(Liang & Xue, 2009)*	PS01 – There is a high probability that IOT systems can cause security breaches.							
	PSO2 – It is likely that the use of IOT systems will lead to misinformation.							
	PSO3 – It is plausible that IOT systems might fail to effectively service clients.							

	PS04 - Use of IOT systems would risk the reputation of the business							
Perceived Severity *(Liang & Xue, 2009)*	PSE01 – If a security breach occurred through in IOT system, the consequences would be severe.							
	PSE02 – Misinformation from IOT systems could have serious repercussions for my job.							
	PSE03 – Failure of IOT systems to effectively service clients can have grave implications for the company.							
Perceived Threat *(Liang & Xue, 2009)*	PT01 – I am worried that IOT systems might increase the risk to my job security.							
	PT02 – I am concerned about the potential threats that IOT systems can bring to our existing systems.							
	PT03 – I perceive the adoption of IOT systems as a threat to the quality of service.							
Adoption Intention	IU01 – Given the opportunity, I plan to use IOT systems in my tasks.							

(Venkatesh et al., 2012)	IU02 – I am willing to integrate IOT systems into my existing workflow							
	IU03 – I could envision adopting IOT systems as a long-term tool for my role.							