N°d'ordre: .....

People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
DJILLALI LIABES UNIVERSITY OF SIDI BEL ABBES
FACULTY OF EXACT SCIENCES
DEPARTMENT OF COMPUTER SCIENCES

Master's Thesis

Domain : Mathematics-computing
Field : Computer science
Speciality : Networks, Systems and Information Security (RSSI)

Par

Mr HADJAZI M.Hisham
Mr SAHRAOUI M.Taher Amine
Mr MLETTA M.Mouncif
Mr DJEBBAR Yehya

WSN's Monitoring Platform With Self-Healing & Post-Quantum cryptography schemes implementation and a Border patrolling and monitoring system.

Defended on 08-07-2023 before the jury :

Dr.	TAIEB BRAHIM Mohammed	UDL-SBA	President of the jury
Dr.	FAHSI Mahmoud	UDL-SBA	Examiner
Dr.	ALAMI Ahmed	UDL-SBA	Incubator representative
Pr.	BOUKLI HACENE Sofiane	UDL-SBA	Supervisor and CATI representative
Pr.	YOUSFATE Abederahmane	UDL-SBA	Co-Supervisor

Academic year : 2022 - 2023

"If you think you understand quantum mechanics, you don't understand quantum mechanics."

Richard Feynman

"If quantum mechanics hasn't profoundly shocked you, you haven't understood it yet."

Niels Bohr

"Quantum computing is a technology that will fundamentally change our world."

Michael Nielsen

"The more success the quantum theory has, the sillier it looks."

Albert Einstein

DJILLALI LIABES UNIVERSITY OF SIDI BEL ABBES
FACULTY OF EXACT SCIENCES
DEPARTMENT OF COMPUTER SCIENCES

Abstract

Faculty of Exact Sciences

Software and Information Systems Engineering

Masters in Software and Information Systems Engineering

WSN's Monitoring Platform
With Self-Healing & Post-Quantum cryptography schemes implementation

by HADJAZI M.Hisham
SAHRAOUI M.Taher Amine
MLETTA M.Mouncif
DJEBBAR Yehya

Wireless Sensor Networks (WSNs) are widely used in various applications, including military operations, environmental monitoring, and smart homes. Security is a critical concern in WSNs, particularly in the presence of quantum computing, which threatens the security of traditional cryptographic systems. The post-quantum cryptography paradigm offers a solution to this issue, but it has several limitations, such as high computational and communication overheads. Self-healing in WSNs is a relatively new and emerging area of research that aims to address these challenges. This thesis proposes a novel approach to enhance the security of WSNs by combining post-quantum cryptography with self-healing techniques. The proposed approach utilizes **CLASSIFIED** cryptography for key exchange, and a **CLASSIFIED** self-healing algorithm for node detection and repair. The effectiveness and performance of the proposed approach are evaluated through extensive simulations comparing it to other self-healing schemes and post-quantum schemes, and the results show that it provides a high level of security with minimal computational and communication overheads. This thesis contributes to the field of self-healing in WSN cryptography by providing a framework for enhancing the security and reliability of WSNs in the presence of quantum computing, and will result in a new start-up called **Quantum Solutions**.

Acknowledgements

First and foremost, we would like to express our deepest gratitude to Allah, the Almighty, for His continuous blessings and guidance throughout our academic journey and the successful completion of our graduation project.

We are immensely grateful to our project supervisors, Professor Sofiane Boucli-Hacene and Professor Abderrahmane Yousfate, for providing us with the opportunity to undertake this project and for their invaluable guidance and support throughout our work. Their expertise, dedication, and motivation have been instrumental in shaping our project and thesis. We are truly privileged and honored to have had the opportunity to work under their supervision. Their mentorship and teachings have not only enhanced our technical skills but also instilled in us a deeper understanding of research methodology and the importance of presenting our work with clarity and precision. We would also like to extend our heartfelt thanks to their wives and families for their acceptance, patience, and support during our discussions and preparations for the project and thesis.

We would also like to express our gratitude to all our professors at UDL University for their commitment to excellence in teaching and for imparting their knowledge and expertise to us. Their passion for their respective fields has inspired us and contributed to our intellectual growth. We are grateful for their guidance, encouragement, and mentorship throughout our academic journey.

Additionally, we would like to thank our classmates and friends who have been a source of support, encouragement, and camaraderie throughout our studies. Their friendship and presence have made our academic experience more enjoyable and memorable.

Finally, we would like to acknowledge the continuous support and love of our families. We express our deepest appreciation to our parents, siblings, and extended family members for their unwavering belief in our abilities, their encouragement during challenging times, and their understanding of the demands of our academic pursuits. Their sacrifices and unconditional love have been our driving force and motivation.

In conclusion, we extend our heartfelt gratitude to Allah, our project supervisors, Professor Sofiane Boucli-Hacene and Professor Abderrahmane Yousfate, all our professors, classmates, friends, and family members who have played a significant role in our academic journey and the completion of our graduation project. Their guidance, support, and encouragement have been invaluable to our success, and we are deeply grateful for their contributions.

Hadjazi Mohammed Hisham

Dedicase

Cher lecteur,

Aujourd'hui, je tiens à dédier ce mémoire à des personnes extraordinaires qui ont joué un rôle essentiel dans ma vie. À mon père bien-aimé, même si tu n'es plus physiquement parmi nous, ta présence continue d'illuminer chaque pas que je fais. Ton amour inconditionnel, ta sagesse et ta force ont été et seront toujours une source d'inspiration pour moi. Ce mémoire est dédié à toi, en reconnaissance de tout ce que tu as fait pour moi.

À ma chère mère, tu as été mon roc, ma confidente et ma meilleure amie. Ta bienveillance et ton soutien indéfectible m'ont permis de surmonter les obstacles avec courage et détermination. Je te dédie ce mémoire en témoignage de ma gratitude éternelle.

À ma petite sœur, tu es ma complice de tous les instants, ma source de joie et de sourires. Ta présence dans ma vie a apporté une touche de légèreté et de bonheur. Ce mémoire est dédié à toi, pour toutes les aventures que nous avons partagées et celles qui nous attendent encore.

À mes amis fidèles, vous êtes les piliers de ma vie. Votre soutien inébranlable, vos encouragements constants et votre amitié précieuse ont été essentiels pour moi tout au long de ce parcours. Je vous dédie ce mémoire avec reconnaissance et affection.

À mes encadreurs, le professeur Boukli et le professeur Yousefate, votre expertise, votre mentorat et vos précieux conseils m'ont guidé sur la voie de la réussite. Ce mémoire est dédié à vous, en témoignage de ma reconnaissance pour votre investissement sans relâche dans mon développement académique.

À mes deux tantes, votre amour et votre bienveillance m'ont entouré d'une chaleur familiale incomparable. Vous avez toujours été présentes pour moi, prêtes à offrir un soutien inconditionnel. Ce mémoire est dédié à vous, en témoignage de ma profonde gratitude pour tout ce que vous avez fait.

Enfin, à mes oncles, vous avez été des modèles inspirants de persévérance et de succès. Vos encouragements et vos conseils avisés ont nourri ma détermination à atteindre mes objectifs. Ce mémoire est dédié à vous, en témoignage de ma reconnaissance pour votre présence bienveillante dans ma vie.

Que cette dédicace soit le reflet de ma profonde gratitude envers ceux qui ont joué un rôle si important dans mon parcours. Leur amour, leur soutien et leur chaleur humaine ont façonné la personne que je suis aujourd'hui.

Avec tout mon amour,

Mletta Mohammed Moncif

شكر

اللهم لك شكر و لك الحمد حتى ترضى و لك الشكر ولك الحمد اذا رضيت ولك الشكر ولك الحمد بعد الرضى. احمد الله واشكره الذي وفقنا في إتمام هذا البحث العلمي و الذي الهمنا العلم و الصحة و العافية و العزيمة على ذلك. فالحمد لله حمدا كثيرا و صلاة و سلام على سيدنا و اول معلم للامة جمعاء آ محمد بن عبد الله آ صلى الله عليه وسلم و آله و صحبه أجمعين في العالمين.

اتقدم بجزيل الشكر و التقدير الى والدي و أهلي لما قدموه لي من دعم مادي و معنوي لاتمام هذه السنوات المباركة بالعلم النافع. كما اتقدم بالشكر و التقدير الى البروفيسور المشرف آ بوكلي حسان سفيان آ و البروفيسور المشرف آ يوسفات عبد الرحمان آ على كل ما قدمه لنا من توجيهات ومعلومات قيمة ساهمت في إثراء موضوع دراستنا في جوانبها المختلفة. كما نتقدم بجزيل الشكر لأعضاء اللجنة المناقشة الموقرة الدكتور آ فحصي محمود آ و الدكتور آ طيب براهيم محمد آ، دون نسيان كل من اساتذتي في التعليم العالي و الثانوي و المتوسط و الابتدائي الذين ساهموا في بناء مستوى ارتقينا به في هذا المجال العلمي الذي يعتبر أساس نهضة وقيام الأمم في زمننا الحالي.

كما أتقدم بالشكر الجزيل لزملائي الكرام الذين ساهموا في تحقيق هذا المشروع المبتكر والذين صبروا ابتغاء الوصول إلى هذه النقطة فكل الشكر والتقدير للاخ الاكبر آ جبار يحيى آ و الاخ الكبير آ حجازي محمد هشام آ و اخي الصغير آ ملاتة محمد منصف آ كما اشكر كل من ساهم في الخفاء في مساعدتي في إتمام جزئي من هذا العمل و المشروع المتمثلين في كل احبابي الذين ساهموا بالدعم المعنوي و كل المطورين الذين ساهموا في تطوير الحزم مستخدمة في برامجنا و كل من ساهم في تطوير بروتوكولات التي قمنا باستغلالها لاختراع هذا المشروع الذي يعتبر اضافة جميلة لهذا العلم الرائع.

صحراوي محمد طاهر أمين

Dedicase

En ce moment de dédicace empreint d'émotion, Je souhaite exprimer ma gratitude à chaque personne, Qui a marqué mon parcours avec une affection, Suivant l'ordre qui résonne dans mon cœur, en fusion.

À mon père, maintenant décédé, source de mon être, Tes valeurs, ta sagesse résonnent à jamais en moi, Tu es l'étoile qui brille dans mon univers si fier, Ton amour continue de guider mes pas, loin des émois.

À ma mère, pilier solide de ma vie, guide éternel, Ton amour infini et ta force sont mes phares, Tu as été présente à chaque instant, d'une façon si réelle, Ton soutien inconditionnel, ma force, mon égide rare.

À mes deux frères, compagnons de mes souvenirs, Vos rires complices et votre présence précieuse, Ont forgé des liens indestructibles, des moments de plaisir, Vous êtes mes alliés, mes complices, mes frères précieux.

À mes trois sœurs, éclats de lumière dans ma vie, Vos sourires radieux et votre affection sincère, Ont coloré mon existence, ont tissé une harmonie, Je vous suis reconnaissant pour votre soutien et votre lumière.

À mes deux précieuses filles, étoiles de ma destinée, Vos sourires illuminent mes jours, mes nuits, Vous êtes la raison de ma fierté, de ma félicité, Mon amour pour vous est infini, mon doux fruit.

À mon épouse bien-aimée, compagne de chaque instant, Tu es ma moitié, mon soutien indéfectible, Ta présence réconfortante est mon enchantement, Dans nos bras, notre amour est incommensurable.

À mes professeurs, gardiens du savoir et de l'esprit, Votre enseignement a nourri ma soif de connaissance, Grâce à vous, mes horizons se sont élargis, infinis, Je vous suis reconnaissant pour votre bienveillance immense.

À mes amis et collègues, présents dans mon quotidien, Vous êtes les piliers qui m'entourent de votre amitié, Nos échanges, nos rires, des moments de soutien, Je suis honoré de partager avec vous cette belle complicité.

Que cette dédicace, tissée de mes émotions les plus pures, Soit une symphonie d'amour, de reconnaissance et de tendresse, À vous tous qui avez marqué ma vie, mes souvenirs, Je vous adresse ma gratitude sincère, mon essence de noblesse.

Que mon mémoire, fruit de cette belle symphonie, Porte en lui les traces de notre parcours partagé, Un hommage à vous, mes proches, ma famille choisie, Le reflet de notre connexion, notre lien sacré.

Avec mes vœux les plus chaleureux pour votre bonheur, Je vous remercie d'avoir enrichi ma vie de votre présence, Que nos chemins continuent de se croiser, en douceur, Dans la joie, l'amour, l'épanouissement, en abondance.

Votre humble dédicataire, porteur de votre amour infini, À jamais relié à vous par les fils invisibles de la vie.

Djebbar Yehya

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Wireless Sensor Networks (WSNs) have become indispensable in modern communication networks, finding applications in environmental monitoring, medical care, smart homes, and military operations. However, the deployment of WSNs in challenging environments exposes them to physical and cyber threats, necessitating the establishment of secure and reliable communication channels to safeguard sensitive data and maintain network integrity. Traditional cryptographic schemes employed in WSNs face vulnerabilities with the advent of quantum computing, demanding the exploration of new post-quantum cryptographic schemes. Furthermore, the dynamic nature of WSNs necessitates an efficient self-healing mechanism to address changing topologies, node failures, and security breaches.

In this context, this thesis aims to investigate a combined post-quantum and self-healing scheme for WSNs, enabling robust and secure communication. Additionally, this research builds upon recent advancements, leveraging our findings to obtain a patent for our innovative solution. Furthermore, two additional patents are currently pending, highlighting the novelty and potential impact of our contributions. Building upon these breakthroughs, we have established a start-up named Quantum Solutions, which specializes in providing comprehensive security services and security kits for households, small businesses, and large-scale government security projects.

The focal point of this thesis revolves around the presentation of our patented prototype, "Border surveillance and monitoring using solar-powered Wireless Sensor Network equipped with 24GHz mmWave sensors and GPS Modules for tracking and ESP-NOW and LoRa protocols for communication." We will provide an in-depth analysis of this prototype, elaborating on its functionality and significance within the realm of border surveillance and monitoring. Additionally, this thesis encompasses the presentation of our business plan, outlining the strategic vision and objectives of Quantum Solutions.

Wireless Sensor Networks (WSNs) have become an essential component in modern communication networks. They are utilized in a variety of applications, including environmental monitoring, medical care and hospitalization, smart homes, and military operations. However, WSNs are often deployed in harsh and unpredictable environments, which make them vulnerable to both physical from forien objects and intended cyber attacks.[VV19] In such scenarios, it is critical to have secure and reliable communication channels that can protect sensitive data and maintain the network's security and integrity.[RP13] One of the key components of any secure WSN is cryptography, and with the rise of quantum computing, traditional cryptographic schemes have become vulnerable, and thus there is a need for new post-quantum cryptographic schemes[Fer20] At the same time, the dynamic nature of WSNs requires an efficient self-healing mechanism to cope with the changing topology, node failures, and attacks. In this context, this thesis explores the use of a combined post-quantum and self-healing scheme in WSNs to provide robust and secure communication.[Asi21][Che+17]

Cryptography is the process of transforming plaintext into ciphertext using cryptographic algorithms. Cryptography in WSNs is typically used for ensuring secure communication between nodes, data confidentiality, and integrity. However, traditional cryptographic methods are vulnerable to a variety of attacks, including brute-force attacks, side-channel attacks, and replay attacks. Moreover, traditional

cryptography techniques do not have the capability to detect and repair the compromised nodes in the network, which makes the entire network vulnerable to attacks.[MS16]

Self-healing in WSN cryptography is a relatively new and emerging area of research that aims to address these challenges. Self-healing refers to the ability of a WSN to detect and repair compromised nodes automatically. The main objective of self-healing is to maintain the network's integrity, confidentiality, and availability in the presence of attacks.[PK22][Guo+18].

The primary aim of self-healing key distribution is to allow group users to retrieve session keys independently, even if they have missed some broadcast messages, without requesting additional transmissions from the group manager (GM). This method is beneficial as it eliminates the need for additional transmission, which can reduce energy consumption. Furthermore, in some scenarios, such as uni-directional broadcast channels from the GM, self-healing key distribution can be the ideal solution. Self-healing key distribution schemes must be energy-efficient, feature short broadcast messages, and efficiently add and revoke users. However, there is a challenge in managing the balance between providing adequate security and conserving scarce resources, particularly energy, which is critical for wireless network operations. Over the past decade, numerous self-healing key distribution schemes have been proposed to establish a group key among dynamic users over an unreliable or lossy network. This paper presents a comprehensive survey of the state-of-the-art in self-healing key distribution. It first clarifies the security requirements for the self-healing key distribution scheme in specific application environments. Next, the paper classifies self-healing key distribution schemes according to different cryptographic primitives and provides insights into their features and goals. Moreover, it considers various problems related to the robustness of self-healing key distribution schemes, such as authentication on broadcast messages, sponsorization, and mutual-healing.[Tia+11][MO15]

Post-quantum cryptography is a new area of research that focuses on developing cryptographic schemes that are secure against quantum attacks. This is achieved by designing schemes that rely on mathematical problems that are hard to solve even for quantum computers. For example, lattice-based cryptography and code-based cryptography are two popular post-quantum cryptographic schemes.[Asi21] In recent years, there has been a growing interest in post-quantum cryptography due to the emergence of quantum computers, which are expected to break the security of many traditional cryptographic schemes.[Pre18]

However, merely replacing traditional cryptographic schemes with post-quantum ones is not sufficient to provide security in WSNs, which have a dynamic topology and are prone to node failures and attacks. To cope with these issues, a self-healing mechanism is needed, which enables the network to recover from such situations without requiring external intervention.[Fer20] Self-healing schemes in WSNs have been designed based on several principles such as redundancy, cooperation, and fault tolerance. These mechanisms help to ensure that the network continues to function even in the presence of node failures and attacks.[Asi21][Fer20]

In conclusion, this thesis proposes a pioneering combined post-quantum and self-healing scheme for WSNs, ensuring secure and robust communication. Leveraging our patented prototype and impending patents, we have established Quantum Solutions, a start-up aimed at providing security services and security kits to diverse

clientele. Through this thesis, we present our patented prototype in detail, highlighting its application in border surveillance and monitoring. Additionally, we outline our business plan, showcasing the strategic vision and objectives of Quantum Solutions. By designing, analyzing, and evaluating the proposed scheme in terms of security and performance, this research aims to contribute to the development of secure cryptographic schemes resilient against classic and quantum attacks, while incorporating a self-healing mechanism to enhance network resilience and effectiveness.

Chapter 2

State of the art

2.1 Self-Healing

With the many uses for wireless sensors in different fields, they need a suitable self-managing method to work because they are challenging to maintain in the long run against errors and faults. The development of wireless sensors has facilitated new pervasive applications, from health care to monitoring wildlife in urban and isolated areas, taking into consideration the limited resources at hand.[BS] so they need a flexible way of adapting, which means that recovery from fault is critical. accuracy is reading is also vital,nodes susceptible to faults from overheating, chemical fouling of sensors, low battery levels, or physical damage resulting in deterioration of sensor accuracy over time. Furthermore, wireless links may have variable as well as asymmetric quality-related works.,the key layers of sensors :Solutions or approaches can be classified as centralized or distributed (in multi-hop) neighborhood collaboration, and in centralized settings we have two monitoring models: active and passive. MANNA is a centralized framework with an active model).Sympathy(centralized framework with passive model).The dissemination of binary images in the network is implied by network reconfiguration.which increases the bandwidth and rebooting time, Maté uses a lightweight virtual machine. TinsOS WITH ADJUSTABLE INSTRUCTIONS Scripts are smaller than binary images, which explains Mate's approach to reconfiguration.[BS]

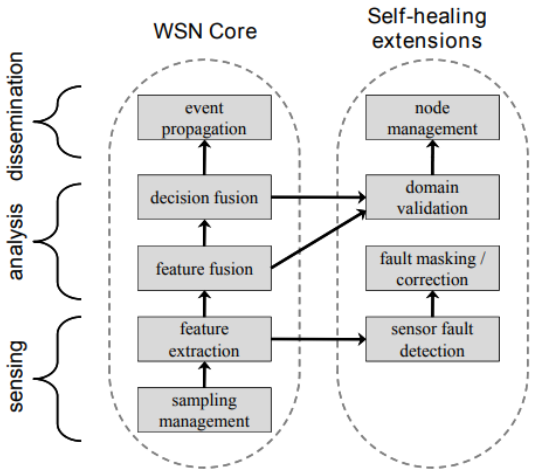


Fig. 1. Functional architecture of a self-healing wireless sensor network

FIGURE 2.1: Functional architecture of a self-healing wireless sensor network.[BS]

In this thesis, Themistoklis and Morris outline the conception and construction of a self-healing platform, which will simplify the creation of pervasive applications and, more especially, the sensing portion of those applications by delivering accurate sensor data; a policy-driven platform that describes missions in terms of high-level goals to make network administration and adaptability easier; a framework for data and functional defect detection that is scalable in terms of resources; and research into statistical data-fault detection techniques that can function in networks with diverse sensor and node types and require less sensor redundancy.[BS]

Biming Tian et all provided an in-depth examination of the state-of-the-art in the

subject of self-healing key distribution. By enabling group users to recover session keys without requesting further transmissions from the group manager, self-healing key distribution aims to save energy.[Tia+11]

The challenges of balancing security and resource saving, notably energy conservation, are covered in the study along with the required characteristics of self-healing key distribution methods. Based on several cryptographic primitives, the study gives a taxonomy of self-healing key distribution methods and highlights each one's salient characteristics and objectives.[Tia+11] Furthermore, The effectiveness of several methods is compared to Polynomial secret-sharing-based self-healing key distribution schemes, Vector space secret-sharing-based self-healing key distribution schemes, SDR-based self-healing key distribution schemes, Hash chain-based self-healing key distribution schemes and Bilinear pairing-based self-healing key distribution schemes. In addition, The study is highlighting their parallels and contrasts. Sponsorization, mutual healing, and authentication are further concepts covered by the authors that can be employed to increase the resilience of these schemes.[Tia+11]

In their conclusion, they identify any unresolved issues in the subject. They covered that ensuring forward and backward secrecy in self-healing key distribution schemes is very important. The authors note that secure and efficient key revocation for self-healing key distribution schemes based on bilinear pairing continues to be a challenge. They also note that finding the best balance between bandwidth efficiency and robustness against link loss while taking power consumption into account is still difficult. Finally, they clarify that is no optimal solution and that the choice of the best schemes relies on the domain of application and the priority of some criteria like computation, complexity, and resilience.[Tia+11]

The practicality of self-healing group key distribution methods for resource constrained WSNs is examined in this work, and it is demonstrated that none of the existing schemes are appropriate for large-scale and resource-restricted WSNs. The foundation of secure group communication for massive groups in wireless sensor networks is self-healing key distribution (WSNs).[Wan11]

In order for them to recover the group key without needing further transmission from the group manager, it entails broadcasting packets that are only usable for trusted members. Reduced network traffic, less work for the group manager, and a lower danger of user exposure due to traffic analysis are all advantages of such an approach. Revocation Polynomial based Self-healing Key Distribution (RP-SKD) schemes with collusion resistance properties are the most optimal in terms of storage and communication overheads for highly critical applications such as military applications.[Wan11] However, with the maximum broadcast packet size of 1KB, the RP-SKD schemes are impractical for implementation in real-world applications for large-scale and resource-constricted WSNs., 2.1 summarizes the performance comparison results of the HC-SKD schemes and 2.2 displays the possible tradeoff between m and t for different schemes given the maximum packet size of broadcasts 1KB.

TABLE 2.1: PERFORMANCE COMPARISONS RESULTS OF DIFFERENT RP-SKD SCHEMES IN THE J-TH SESSION.[Wan11]

Schemes	Storage overhead	Communication overhead	Forward security	Backward security	collusion attack Resistance
Construction 3 of [2] J.Staddon et al 2002	(m-j+1)2 logq	(mt2 +2m+m+t) logq	Yes	Yes	Yes
Scheme 2 of [3]] C.Blundo et al 2003	(m-j+1) logq	(jt2 +jt) logq	Yes	Yes	Yes
Scheme 3 of [4]] D.Liu,et al 2003	2(m-j+1) logq	[(m+j+1)t+(m+1)] logq	Yes	Yes	Yes
Scheme 3 of [5] C.Blundo et al 2004	(m-j+1) logq	2t+1)j logq	Yes	Yes	Yes
Scheme 2 of [6] D.Hong et al 2005	(m-j+1) logq	(t+1)jlogq	Yes	Yes	Yes

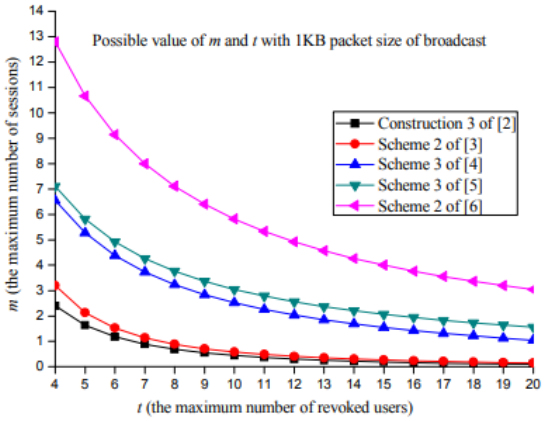


Fig. 1. Possible values of m and t for different RP-SKD schemes with 1KB packet size of broadcast

FIGURE 2.2: Possible values of m and t for different RP-SKD schemes with 1KB packet size of broadcast.[Wan11]

2.2 shows that the schemes [7, 9, 10] are highly efficient, but have a serious security problem and can't guarantee backward security. 2.3 also shows that the value of m and t is too small to be applied in real-world applications of WSNs, making them unsuitable for large-scale WSNs.[Wan11]

TABLE 2.2: PERFORMANCE COMPARISONS RESULTS OF DIFFERENT HC-SKD SCHEMES IN THE J-TH SESSION.[Wan11]

Schemes	Storage overhead	Communication overhead	Forward security	Backward security	Collusion attack resistance
Scheme of [7] R. Dutta et al 2007	3 logq	(t+j+1) logq	No	No	No
Scheme 2 of [8] [8] R.Dutta 2007	(t+3) logq	(t+1)j logq	Yes	Yes	No
Scheme of [9] R. Dutta 2007	(t+3) logq	(t+1+j) logq	Yes	No	No
Scheme of [10] R.Dutta 2008	(m-j+1) log q	(t+1+j) logq	Yes	No	No
Scheme of [11] W. Du, M.X. He and X. Li.,2009	3 logq	(3t+j+2) logq	Yes	Yes	NO

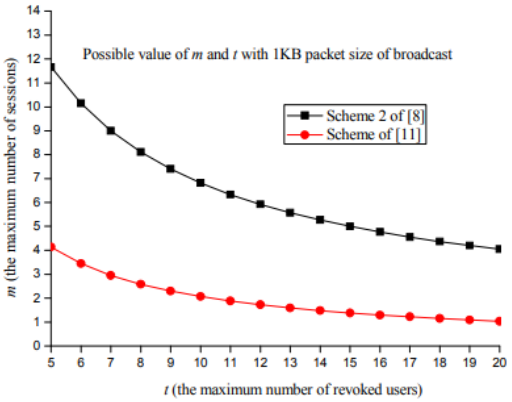


Fig. 2. Possible values of m and t for different HC-SKD schemes with 1KB packet size of broadcast

FIGURE 2.3: Possible value of m and t with 1KB packet size of broadcast.[Wan11]

.2.3 and 2.4 show that the AP-SKD schemes are not suitable for large-scale WSNs. The maximum number of session m and the maximum number of active group members |Gj| are too small to be applied in practical application of large-scale schemes, such as when the maximum number is 10 and 51, respectively.[Wan11]

TABLE 2.3: PERFORMANCE COMPARISONS RESULTS OF DIFFERENT AP-SKD SCHEMES IN THE J-TH SESSION.[Wan11]

Schemes	Storage overhead	Communication overhead	Forward security	Backward security	collusion attack Resistance
Scheme of [12] X.K. Zou and Y.S. Dai 2006	3 logq	max{(3t+1)j, [(|Gj+2t+2)j]} logq	Yes	Yes	Yes
Scheme of [13] B.M. Tian	3 logq	max{(t+1)j, (|Gj+2)j} log	Yes	Yes	Yes
Scheme 1 of [14] Q.Y. Xu and M.X. He 2009	4 logq	max{(t+j+1), (|Gj|+j+2)} logq	Yes	No	No
Scheme 2 of [14] Q.Y. Xu and M.X. He 2009	5 logq	max{(t+1), (|Gj|+2)} logq	Yes	No	No

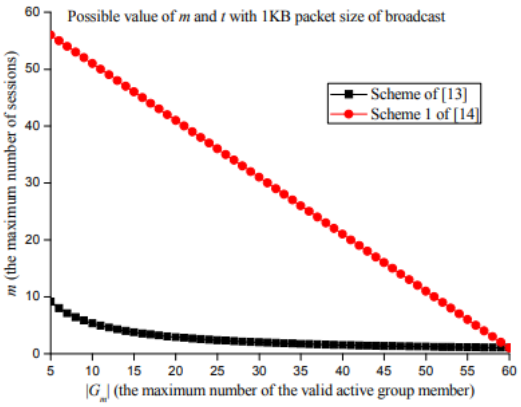


Fig. 3. Possible values of m and t for different AP-SKD schemes with 1KB packet size of broadcast

FIGURE 2.4: Possible values of m and t for different HC-SKD schemes with 1KB packet size of broadcast.[Wan11]

In Qiuhua Wang paper he examines the practicality of three types of self-healing

group key distribution schemes for resource-constricted WSNs. It concludes that none of the existing schemes is suitable for large-scale WSNs in real-world applications.[Wan11] To recover the lost session key from the current broadcast packet, the group manager must include all of the previous broadcasted messages in the current session key updating broadcast packet, resulting in large redundancy. It is important to consider the composition of different pieces of the broadcast messages carefully when designing new schemes. In future work, we will try to develop new schemes suitable for specific applications of WSNs with better security and efficiency.

2.2 Post-Quantum

As the popularity of the Internet of Things (IoT) grows, there is a growing concern over the security of these devices, particularly when it comes to public-key encryption systems. These systems are crucial for ensuring the security of non-secure communication channels, such as those used for financial transactions, exchanging emails, accessing websites, and more. The threat of quantum computing is particularly concerning for these systems, as it poses a threat to long-term IoT device security.[Fer20]

IoT devices are typically battery-dependent and have limited resources in terms of computational power and memory, making it difficult to implement complex encryption algorithms. This is particularly challenging for applications that require data privacy to be preserved for the long term, such as IoT applications for defense, mission-critical scenarios, or smart healthcare.[Fer20]

To address this challenge, researchers are developing post-quantum IoT systems. These systems are designed to be protected from known quantum computing attacks, and they rely on specific energy-efficient and lightweight algorithms that can be implemented on resource-constrained IoT devices. This article provides a survey of post-quantum IoT systems, including the main post-quantum cryptosystems and initiatives, the most relevant IoT architectures and challenges, and the expected future trends.[Fer20]

One of the most significant challenges facing post-quantum IoT systems is the need to standardize protocols and develop appropriate architectures to provide services to IoT devices. As the number of IoT devices grows, there will be a need to ensure that these devices can communicate with each other securely and that they can be integrated with other systems securely and efficiently.[Fer20]

In terms of cryptography, several post-quantum public-key cryptosystems can be used for potential IoT applications. These include schemes based on lattice-based cryptography, code-based cryptography, multivariate cryptography, and hash-based cryptography. Each of these schemes has its strengths and weaknesses, and the choice of the scheme will depend on the specific requirements of the application.[Fer20]

Quantum computers can quickly solve complex classical cryptography equations,

making traditional encryption vulnerable to hacking.[Asi21] As a result, new strategies for data encryption in the post-quantum era are being developed, with lattice-based cryptography (LBC) being a promising approach due to its resistance to quantum attacks and high performance.[Asi21] However, implementing LBC in energy-restricted Internet-of-Things devices requires careful study of lightweight lattice-based cryptography (LW-LBC). This survey aims to provide information on mathematical facts, real-time implementation, hardware architecture, open problems, attack vectors, and the significance of LBC for IoT networks.[Asi21]

As implementing a secure Quantum Key Distribution (QKD) system has several drawbacks, there is a growing interest in developing classic, non-quantum cryptographic algorithms that can operate in current real-time infrastructures. These quantum-robust algorithms, known as Post-Quantum Cryptography (PQC) algorithms, are expected to remain secure even after the availability of large-scale quantum computing machines. To be effective, modern cryptosystems must be combined with existing protocols like transport layer security, and their design should consider factors such as key and signature size, encryption/decryption time, and amount of traffic needed for encryption, decryption, or signature transmission.[Asi21]

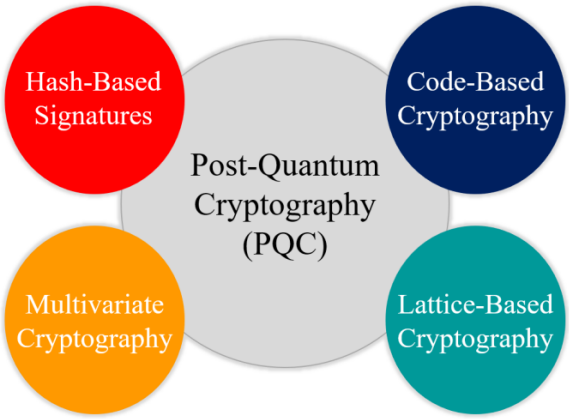


FIGURE 2.5: Possible values of m and t for different AP-SKD schemes with 1KB packet size of broadcast.[Asi21]

Most cryptographic structures can be categorized into four families, which are lattice-based, multivariate, hash-based (signatures only), and code-based. Some algorithms may not be practical for use in IoT networks due to latency and integration concerns.[Asi21] Designers of post-quantum cryptosystems must consider these attributes for IoT use cases. Post-quantum protocols offer a range of primitives to address challenges in implementing across different computing platforms and use cases, providing security beyond classical cryptography. These algorithms are being implemented in next-generation networks and mission-critical infrastructures, such as energy, medical, and space exploration.[Asi21]

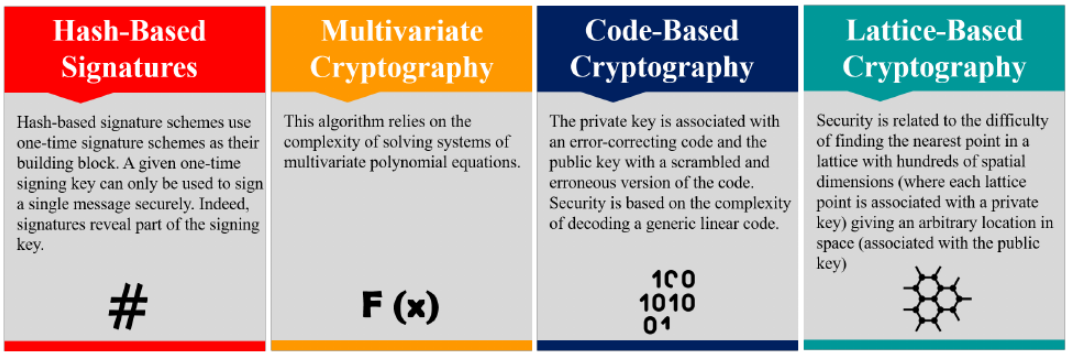


FIGURE 2.6: Implementation methods of four basic quantum secure algorithms.[Asi21]

This survey examines the feasibility of using post-quantum cryptography in resource-limited devices, like IoT. We compare various post-quantum key exchange schemes based on their memory usage, computational time, and clock cycle counts on hardware platforms. Lattice-based cryptography (LBC) is a promising alternative for IoT networks to resist quantum computation. Versatile processors, such as FPGAs, ASICs, and Raspberry Pi, allow low-power devices to handle the hardest quantum encryption systems. Researchers are adapting advanced hardware designs based on number theoretical transformation (NTT) for the lightweight implementation of LBC, with an updated NTT that separates data from vectors and allocates memory with smaller footprints for reduced energy consumption, while maintaining desired throughput and security. LBC's scalability and flexibility make it the primary candidate for post-quantum IoT security.[Asi21]

TABLE 2.4: Summary of the widely deployed classical cryptographic systems and their security levels against the best pre-quantum and post-quantum attacks known.[Wan11]

Name	Function	Pre-Quantum Security Level	Post-Quantum Security Level
Symmetric Cryptography (Private Key)			
AES-128	Block Cipher	128	64 (Grover)
AES-256	Block Cipher	256	128(Grover)
SALSA-20	Stream Cipher	256	128(Grover)
GMAC	MAC	128	128(Grover)
POLY-1305	MAC	128	128(Grover)
SHA-256	Hash Function	256	128(Grover)
SHA-3	Hash Function	256	128(Grover)
Asymmetric Cryptography (Public Key)			
RSA-3072	Encryption	128	Broken (Shor)
RSA-3072	Signature	128	Broken (Shor)
DH-3072	Key Exchange	128	Broken (Shor)
DSA-3072	Signaure	128	Broken (Shor)
256-bit ECDH	Key Exchange	128	Broken (Shor)
256-bit ECDSA	Signature	128	Broken (Shor)

Overall, the future of post-quantum IoT systems is promising, with researchers working to develop new algorithms and architectures that can provide robust and reliable security for IoT devices. As the number of IoT devices continues to grow, there will be a growing need for these systems to ensure that the data transmitted between these devices is secure and protected from potential quantum computing attacks.

By following the guidelines provided in this article, future IoT developers can create quantum-resistant solutions that can help to secure the future of IoT.[Fer20]

2.3 Application: Border Surveillance and Monitoring

The idea of using sensors scattered over a peice of land to monitor border crossings is not new, and there are many implementations of it over the Previous years. however our implementation builds over what other people have already accomplished and we add what we think are shortcommings in thier ideas in order to further improve the concept and make it a more feasible solution.

2.3.1 "IN202141013947 - DESIGN AND IMPLEMENTATION OF SCALABLE WIRELESS SENSOR NETWORK BEYOND WI-FI" by Wilfred J Vaz and Mrinal Sarvagya

Wilfred J Vaz and **Mrinal Sarvagya** from India with the **"DESIGN AND IMPLEMENTATION OF SCALABLE WIRELESS SENSOR NETWORK BEYOND WI-FI"**[WM21] which is a system that uses ESP32 microcontroller equipped with LoRa modem and GPS module and I2C interfaced-multiplexed sensors, they used the ESP32 as nodes for their Wireless sensor network.[WM21] Their system was used for three tasks first to monitor soil and help improve smart farming, second to monitor pollution inside cities, and finally to do environmental monitoring to help with natural disasters.[WM21]

As our application is to monitor different border crossing, we will be using a different type of monitoring sensors which are the "24GHz mmWave sensors", second since our sensors will be closer to each other we will be using the "ESP-NOW" protocol to for communication between nodes, this will drastically improve power consumption as it consumes much less power than LoRa, and we will only use LoRa protocol when our nodes start to die and the need for a long-distance form of communication is required to maintain the network.

Our system also uses solar power to charge the batteries, making them lasting much longer.

2.3.2 "CN206629277 - WSN MONITORING SYSTEM" by LIU JIANYUAN, PAN ZHONGMING,ZHANG HENG, ZHANG ZHUOHANG, and WANG HUITANG

Here **LIU JIANYUAN**, **PAN ZHONGMING**, **ZHANG HENG**, **ZHANG ZHUOHANG**, and **WANG HUITANG** from Chaina with their **"WSN MONITORING SYSTEM"** have made a very complicated system that consists of many types of sensors in one node the sensors are a magnetic sensor module, a sound array sensor module, a vibration sensor module, a Doppler microwave sensor module, and an infrared sensor module.[JIA+17] now the use of all these sensors will complicate the system and make it more expensive not to mention the power consumption.[JIA+17] and as they have mentioned that the system is battery powered they will need a larger battery making their system not suitable for border control, and much more suitable for monitoring smaller scale areas like airports and Buildings.[JIA+17]

Our system is more of a disposable system that can last for years without the need for any field maintenance. this is due to using fewer sensors to save resources. and

using solar power to charge the batteries. which means they can work for the lifetime of their components.

Another difference is the use of ESP-NOW protocol for all short-distance communications and only use LoRa protocol when it is impossible for the nodes to communicate using ESP-NOW due to large distance or the malfunction of nearby nodes.

Our system doesn't use CCTV cameras as it is more suitable for rural areas where CCTV cameras will make the system more complicated for very little benefit.

2.3.3 "IN202211052636 - SYSTEM FOR CONTROLLING AND MONITORING SOLAR-POWERED, WIRELESS SENSOR NODES IN PRECISION AGRICULTURE" by Dr. Pravin P Patil, Dr. Rupa Khanna Malhotra, Dr. Savita, and Dr. Bhasker Pant

Dr.Patil, **Dr.Malhotra**, **Dr. Savita**, and **Dr.Pant**describes systems and methods related to wireless sensor networks that harvest ambient energy for intelligent agricultural control.[Pat+]

They have made Solar energy harvesting wireless sensor networks (SEH-WSNs) added to WSN.[Pat+]

Our system is used for a different application as we are trying to monitor the movements of land objects in largely rural areas.

Not to mention the differences in communication protocols as we are not using ZigBee in our system.

2.3.4 "IN201921024557 - INTRUDER MOTION DETECTION SYSTEM" by Mrunal Mohan Khedkar and Gajendra Mahadeorao Asutkar

Khedkar and **Asutkar** have made a wireless intruder motion detection system using digital cameras. The system can detect intruders, animals, or objects through a motion detection technique that turns on the camera when movement is detected. The camera can capture videos during both day and night, and the system can transmit the captured videos to a base station or server wirelessly.[KA]

The system uses solar-powered batteries to keep the system alive, and the cameras only turn on once movement is detected to save energy.[KA]

Our system doesn't use cameras for monitoring. we use microwave sensors to detect the movement that can be analyzed at the base to identify the object solely on sensor data.

Our system can also be deployed on very large areas that stretch for hundreds of kilometers.

Chapter 3

Software: WSN nodes

3.1 AD-HOC networks

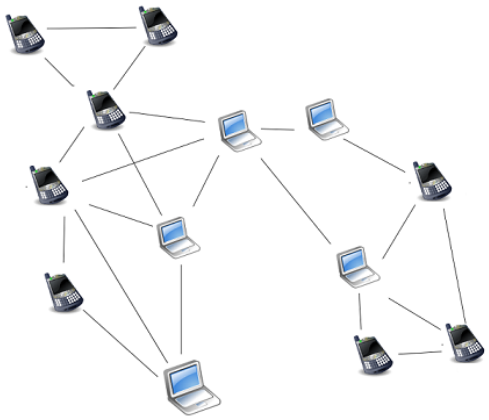


FIGURE 3.1: AD-HOC network

The term "Ad Hoc" (temporary or when needed) in the phrase Ad-Hoc network itself tells us the meaning of Ad-hoc network. It means a network that is temporary or set up immediately when needed for random temporary access. So an Ad Hoc network is a group of "locally-available" devices that can connect and talk to each other directly whenever needed without the need of any router or server. This network breaks when the connected devices go out of the network i.e. disconnect from the network, so the network breaks with the closing of the Ad-Hoc connection. Mobile Ad Hoc Network or MANET can be explained as a transportable network that can be created and implemented as and when required based on the requirement set on the scenario, and hence the name has 'ad hoc' in it. It is extensively chosen due to its significant characteristics like the auto-configuring ability, ability to function as a cordless or wireless network, self-diagnosing capability when an issue arises in connectivity, etc.

These networks use device-to-device communication instead of using the hardware used in traditional networks. These networks skip or avoid the hardware of a traditional network. These are local area networks (LANs) that allow devices in a range of each other to connect wirelessly, instead of using a fixed server or router to act as a central hub for connected devices.

3.1.1 What are Mobile Ad Hoc Networks?

We just need to add the term "mobile" with them which will lead us to MANET i.e. Mobile Ad-hoc Networks. Thus, MANETs can be defined as the Ad Hoc networks which use mobile devices i.e. movable wireless devices (laptops, smartphones, iPads, etc.) as their nodes of connection and communication.

An example of MANET is a VANET (Vehicular Ad-hoc Network) in which communication devices are installed inside vehicles to share data of traffic among the cars. Also, wireless sensor networks are an example of MANETs.

3.1.1.1 MANETs Features

- Work as an individual exclusively or in a group as a part of a huge network like the internet.

- No access point or hardware.
- Direct and autonomous communication between mobile devices.
- The devices search each other themselves and start communicating.
- If a node is far away, then the nodes between the source and destination nodes act as routers and transfer data one by one to make it reach the destination node.
- The devices can join or leave the network anytime i.e. dynamic node addition or removal happens.
- The devices have their own backups of energy like they have batteries for power.

3.1.1.2 Characteristics of Mobile Ad-Hoc Networks

- No Centralized Control: The working is completely dependent on the behavior and support of the participating devices.
- Random Change of Devices: Devices keep on coming and leaving the network at a huge pace and voluntarily.
- Frequent changes in network topology i.e. the arrangement of devices in the network
- Less Security: These networks have bigger threats than on wired networks.
- Limited Bandwidth: These networks have very little capacity and range of data transmission.
- Low resources such as memory power, battery power, backups, etc.

3.2 WSN "Mesh network topology"

Mesh networks are a sort of ad-hoc network that let gadgets talk to one another without the need for a centralized infrastructure. These networks are made up of interconnected devices that have a mesh architecture. Every component of the network serves as a router, passing data packets to other components. Mesh networks may route data around broken nodes, making them extremely robust to network outages. Mesh networks can also emerge on their own, without any design or configuration beforehand. Mesh networks can be challenging to manage and protect, have a restricted range and bandwidth, and have other drawbacks. Mesh networks are a viable replacement for conventional wired and wireless networks overall, but more study and development are required to properly tap into their potential. What possible uses for mesh networks exist, and how might they be applied to enhance current infrastructures? Mesh networks have a wide range of uses, such as in smart cities, military operations, and disaster assistance. Mesh networks can be employed in disaster relief to offer connectivity and communication in locations where conventional infrastructure has been devastated. Mesh networks can be utilized in military operations to give personnel on the ground safe and dependable communications. Mesh networks may be used in smart cities to link smart devices like sensors and IoT gadgets.

Mesh networks may be utilized to upgrade current infrastructures as well. Mesh networks, for instance, can be used to offer backup connections in places where conventional infrastructure is problematic, such as isolated or rural regions.

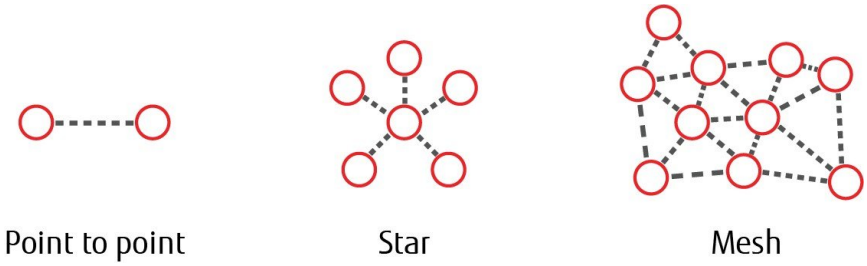


FIGURE 3.2: Diffrent network topologies

3.2.1 Full Mesh and Partial Mesh Topologies:

There are two types of mesh network topologies: full mesh and partial mesh. In a full mesh topology, every node forms direct connections with every other node, but in a partial mesh topology, only a select few nodes have direct connections with one another. In partial mesh networks, certain nodes can also have to traverse through intermediary nodes in order to reach their intended locations. Depending on the individual requirements and factors to be taken into account, complete mesh or partial mesh configurations can be set up in both wired and wireless mesh networks.

3.2.2 Logical and Physical Topologies:

Due to the ability to transport data between users, most networks may appear to be completely meshed, but with mesh networks, it's important to distinguish between logical and physical topologies. Physical topology focuses on the actual configuration of nodes and connections, whereas logical topology relates to the network protocols enabling complete interconnection. Notably, complete mesh networks are frequently used in data center fabrics, which guarantee high-bandwidth connection, and partial mesh or tree topologies are frequently used in wide-area networks (WANs).

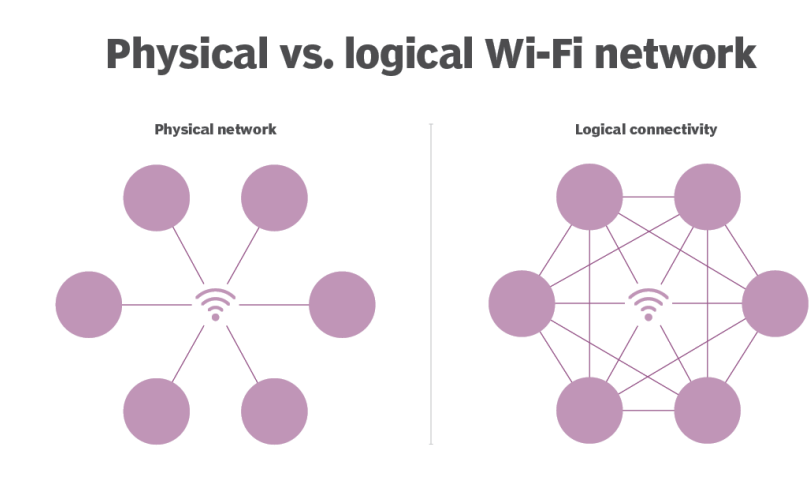


FIGURE 3.3: Physiscal vs Logical topologies

3.2.3 Mesh Network Functionality:

Mesh network nodes have software that controls how they interact with the network and how they handle data. Routing solutions require constant connections and self-healing algorithms to adjust to broken pathways since messages hop between nodes before reaching their destination. As an alternative, flooding strategies include sending data from a section of the network's nodes to the whole network, with each node receiving some of the data. A protocol chooses the senders for each data transmission in order to maximize throughput.

3.2.4 Practical Applications of Mesh Networks:

Mesh networks are useful in a variety of settings, from small home networks to huge corporations. In larger places, their advantages become very noticeable. Mesh networks provide a direct connection between devices without relying on the internet and make it possible to share internet access across several devices. Home monitoring, industrial and medical monitoring and control, security systems, and communication for public services are a few examples of use cases. For instance, a mesh architecture with several sensor nodes may effectively monitor a large region.

3.2.5 Benefits of Mesh Networks:

Mesh networks have a number of advantages over other network configurations, including improved stability due to the absence of single points of failure, increased range with fewer dead spots, direct communication between nodes without the need for a central access point, lower power requirements for individual nodes, improved security due to the ease with which attacked nodes can be replaced, and simplified topology that requires less infrastructure.

3.2.6 Drawbacks of Mesh Networks:

Mesh networks provide a lot of benefits, but they also have certain disadvantages. Costs are higher compared to traditional network setups, there may be difficulties scaling the network size based on the number of nodes needed, managing and troubleshooting a large number of nodes is more difficult, there may be latency issues with low-power wide area networks, and mesh network deployment may be challenging due to power consumption issues.

The features, functionality, and real-world uses of full mesh and partial mesh network topologies have been clarified by this scholarly investigation. Network managers and researchers may choose the best topology for their unique requirements by knowing the advantages and disadvantages of mesh networks. To overcome the difficulties mesh networks face and improve their scalability and effectiveness, further study and technology development are required.

3.2.7 Difference Between Mesh Networks and Traditional Wi-Fi:

A single access point serves as the primary hub for connecting devices in conventional Wi-Fi networks. Devices must seek authorization to connect to this access point since all traffic is directed via it. Mesh networks, in contrast, share the network load across several devices to create a decentralized network. In a mesh network, devices work together to share data amongst clients, reducing the need for

a single access point. Traditional Wi-Fi is appropriate for smaller budgets and locations, whereas mesh networks are especially beneficial for regions with greater coverage.

3.2.8 Comparing Mesh Networks and Traditional Wi-Fi in Business Environments:

Depending on the requirements, business networks can be established utilizing mesh networks or other forms of networks. It is crucial to take into account elements like network design, security, scalability, and performance while putting up a new company network. Organizations may optimize their network infrastructure for effective operations by picking the appropriate network type and creating a thorough network diagram.

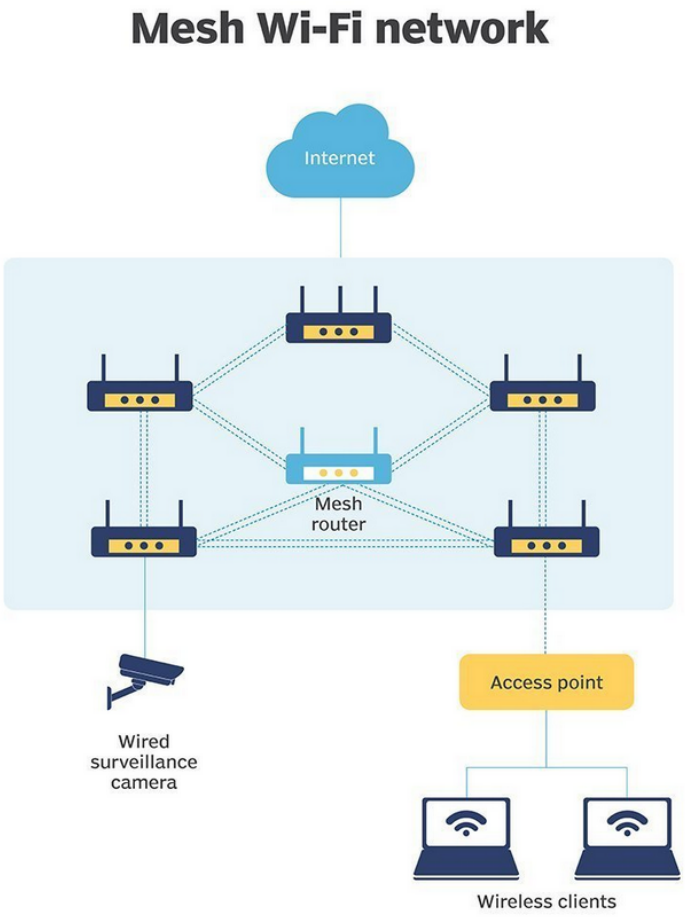


FIGURE 3.4: Mesh wifi network

3.3 ESP-WIFI-MESH

ESP-WIFI-MESH is a networking protocol that utilizes the Wi-Fi protocol as its foundation. It enables the seamless interconnection of multiple devices, referred to as

nodes, across extensive indoor and outdoor areas, forming a unified Wireless Local-Area Network (WLAN). ESP-WIFI-MESH operates in a self-organizing and self-healing manner, allowing the network to be established and maintained independently.

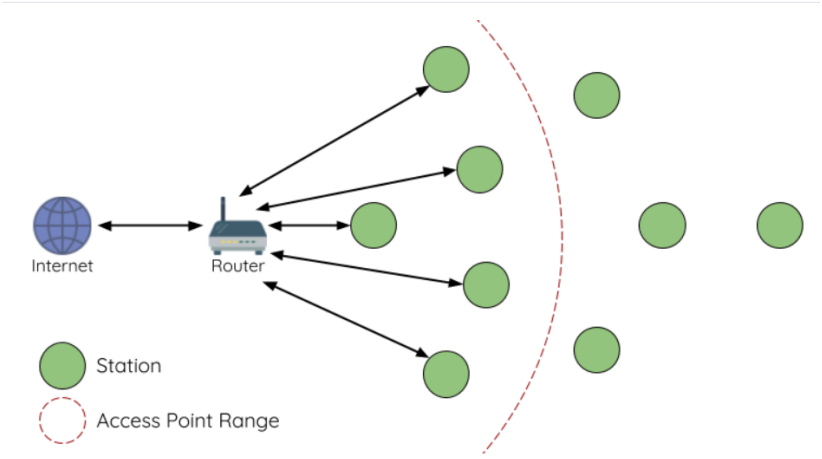


FIGURE 3.5: Traditional Wi-Fi Network Architecture

In a traditional infrastructure Wi-Fi network, there is a hierarchical structure where a central node called the access point (AP) is connected directly to multiple other nodes called stations. The AP acts as a mediator and facilitates the transmission of data between the stations. In some cases, the AP also relays data to and from an external IP network using a router. However, traditional infrastructure Wi-Fi networks have limitations. They have a restricted coverage area since every station must be within range to establish a direct connection with the AP. Additionally, these networks are prone to overload because the maximum number of stations allowed in the network is determined by the AP's capacity.

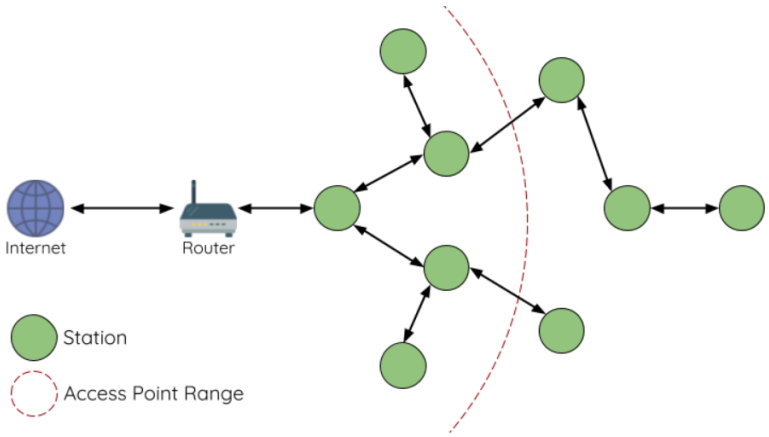


FIGURE 3.6: WIFI-MESH Network Architecture

In contrast to conventional infrastructure Wi-Fi networks, ESP-WIFI-MESH does not require nodes to connect directly to a central node. Nodes in the ESP-WIFI-MESH network are free to link with other nodes, and they all take part in relaying broadcasts from one another. Because nodes may retain interconnectivity without being constrained by the range of a central node, ESP-WIFI-MESH networks can reach

substantially wider coverage regions. Furthermore, since the number of nodes permitted on the network is not restricted by a single central node, ESP-WIFI-MESH networks are less prone to overloading.

3.3.1 Tree Topology

The infrastructure Wi-Fi protocol serves as the basis for ESP-WIFI-MESH, which may be thought of as a networking protocol that combines many separate Wi-Fi networks into a single WLAN. In conventional Wi-Fi networks, stations can only create one connection with an access point (AP) (upstream connection), but an AP can create numerous simultaneous connections with stations (downstream connections). However, ESP-WIFI-MESH adds a special feature where nodes may simultaneously serve as a station and an AP. A node in the ESP-WIFI-MESH may therefore retain a single upstream connection through its station interface while establishing numerous downstream connections through its softAP interface. This property naturally results in a tree network architecture with numerous levels of a parent-child connection and a hierarchical structure.

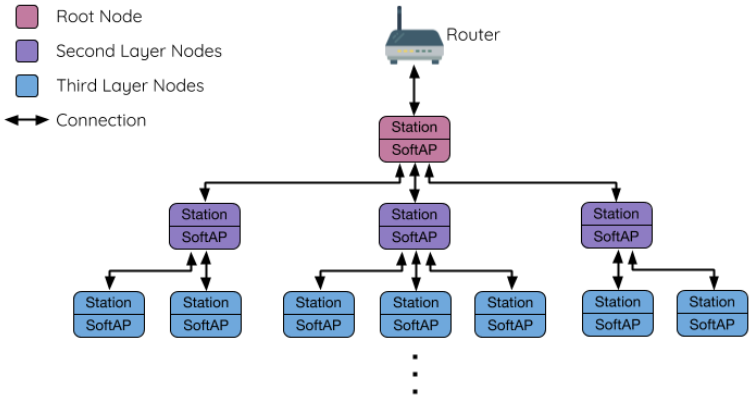


FIGURE 3.7: MESH Tree Topology

As a multi-hop network, ESP-WIFI-MESH enables nodes to send packets over one or more wireless hops to other nodes in the network. This indicates that nodes in the ESP-WIFI-MESH function as relays for other nodes in addition to transmitting their own packets. In an ESP-WIFI-MESH network, communication is feasible between any two nodes as long as there is a practical route between them on the physical layer, accomplished by one or more wireless hops.

It's critical to understand that the maximum number of layers permitted in the network and the maximum number of downstream connections each node may create determine the size of an ESP-WIFI-MESH network.. These variables can be adjusted to set limits on the network's size and capacity.

3.3.2 Node Types

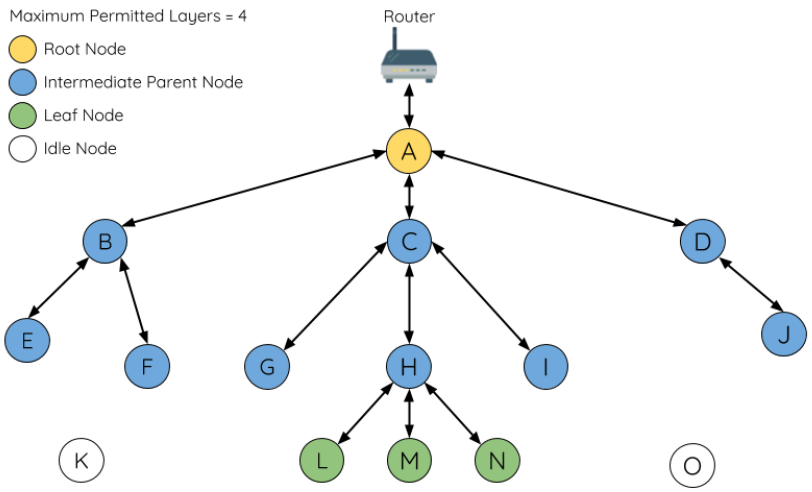


FIGURE 3.8: MESH Node Types

- **Root Node:** The root node, which is placed at the network's top, serves as the interface between the ESP-WIFI-MESH network and an external IP network. It connects to a regular Wi-Fi router and allows packets to be sent between the external IP network and nodes in the ESP-WIFI-MESH network. In an ESP-WIFI-MESH network, only one root node is permitted, and its only upstream connection is with the router. Node A in the diagram symbolizes the root node.
- **Leaf Nodes:** are nodes that are not allowed to have any child nodes or downstream connections. As a result, leaf nodes can only send and receive their own packets and cannot route messages to other nodes. To prohibit the insertion of an extra layer, a node is identified as a leaf node if it is located at the network's maximum permissible layer. Furthermore, because downstream connections need a softAP interface, certain nodes that lack a softAP interface (station-only nodes) are allocated as leaf nodes. Nodes L/M/N in the diagram above are classified as leaf nodes since they are located at the network's maximum allowable layer.
- **Intermediate Parent Nodes:** Intermediate parent nodes are linked nodes that do not function as the root or leaf node. An intermediate parent node must have a single upstream connection (a parent node) but can have zero to many downstream connections (child nodes). As a result, intermediary parent nodes can send, receive, and forward packets from upstream and downstream connections. Nodes B through J in the schematic indicate intermediate parent nodes. Intermediate parent nodes with no downstream connections, such as nodes E/F/G/I/J, differ from leaf nodes in that they may develop downstream connections in the future.
- **Idle Nodes:** Idle nodes are nodes that have yet to join the network. These nodes seek to link upstream to an intermediate parent node or, under certain conditions (as stated in Automatic Root Node Selection), to become the root node. Nodes K and O in the figure above are idle nodes.

Beacon Frames and RSSI Thresholding: If nodes in the ESP-WIFI-MESH network are capable of creating downstream connections (i.e., have a softAP interface), they

will periodically emit Wi-Fi beacon frames. Beacon frames are used to notify neighboring nodes of the existence and status of the transmitting node. Idle nodes listen for beacon frames to create a list of prospective parent nodes with whom they might form an upstream connection. In ESP-WIFI-MESH, the Vendor Information Element is used to store metadata such as node type (root, intermediate parent, leaf, idle), current layer, maximum number of layers permitted in the network, current number of child nodes, and maximum number of downstream connections allowed. The Received Signal Strength Indication (RSSI) of the beacon frames sent out by the putative parent node provides information about the strength of a potential upstream connection. ESP-WIFI-MESH has an RSSI threshold mechanism for beacon packets to stop nodes from creating shaky upstream connections. The transmitting node will be ignored while establishing an upstream connection if a node detects a beacon frame with an RSSI below a predefined threshold.

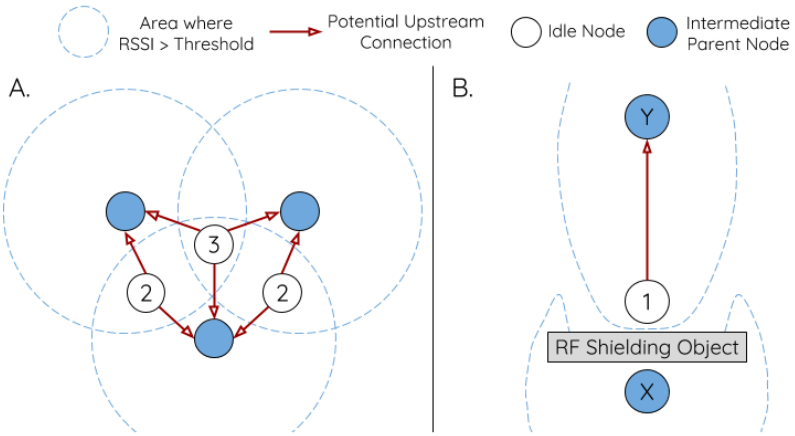


FIGURE 3.9: Effects of RSSI Thresholding

An idle node's access to parent node candidates is affected by the RSSI threshold, as shown in Panel A of the supplied figure. The graphic demonstrates that when the RSSI threshold drops, there are more possible parent nodes. This implies that a node that is idle may take into account additional nodes as prospective upstream connections.

The illustration's Panel B shows how an item that causes RF shielding might reduce a possible parent node's RSSI. The region where node X's RSSI exceeds the threshold is greatly decreased as a result of the RF shielding item. Therefore, although being nearby physically, node X is ignored by the idle node. Instead, the idle node creates an upstream connection with node Y, which is farther away but has a greater RSSI.

Note that nodes continue to receive all beacon packets at the MAC layer. An ESP-WIFI-MESH feature called the RSSI threshold filters out received beacon frames below a set threshold.

3.3.2.1 Preferred Parent Node:

When an idle node has multiple potential parent nodes, it selects a preferred parent node based on specific criteria. These criteria include:

1. The layer on which the parent node candidate is situated.

2. The number of downstream connections (child nodes) the parent node candidate currently has.

The following guidelines are always used to choose the optimal parent node:

- The parent node candidate that is situated on the network's shallowest layer, including the root node, is given preference. When creating upstream connections, this method reduces the overall number of layers in the ESP-WIFI-MESH network. For instance, the second layer node will always be chosen if there are parent node candidates on the second and third tiers.
- The candidate with the fewest child nodes receives preference if there are many parent node candidates inside the same layer. This criterion makes sure that nodes on the same layer have an equal distribution of downstream connections.

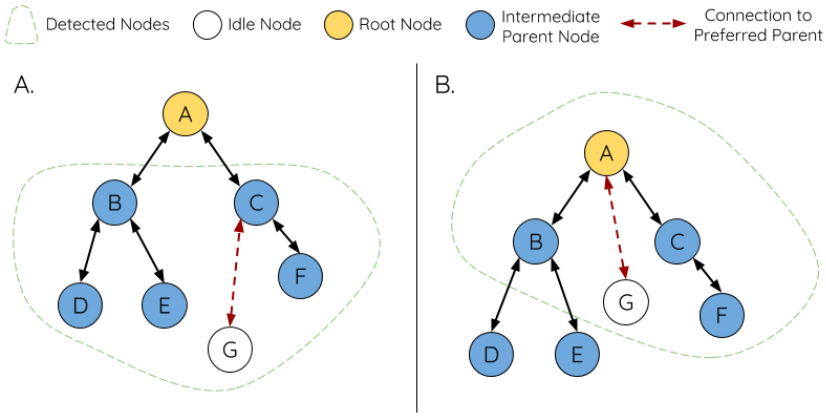


FIGURE 3.10: Preferred Parent Node Selection

An example of how an idle node G chooses a preferred parent node from among five parent node candidates—B, C, D, E, and F—is shown in Panel A of the supplied figure. Nodes B and C are chosen because they are second-layer nodes since the selection procedure gives the highest priority to nodes on the shallowest layer. Since node C has fewer downstream connections (child nodes) than node B, it is finally selected as the preferred parent node.

The illustration's panel B depicts a situation in which the root node is in the vicinity of idle node G. This indicates that the beacon frames that node G receives from the root node surpass the RSSI threshold. When there are numerous parent node candidates, the root node, which is always the shallowest node in an ESP-WIFI-MESH network, is automatically chosen as the preferred parent node.

It should be noted that users can also create their own algorithms for choosing a preferred parent node or have a node only connect to a certain parent node. Refer to the "Mesh Manual Networking Example" in the ESP-WIFI-MESH manual for further details.

3.3.3 Routing Tables:

Each node in an ESP-WIFI-MESH network has its own routing table that is used to route ESP-WIFI-MESH packets (also known as ESP-WIFI-MESH packets) correctly to the desired destination node. A given node's routing table contains the MAC

addresses of every node in its subnetwork, including that node's own MAC address. The routing table is internally partitioned into a number of subtables, each of which corresponds to the child node's subnetwork.

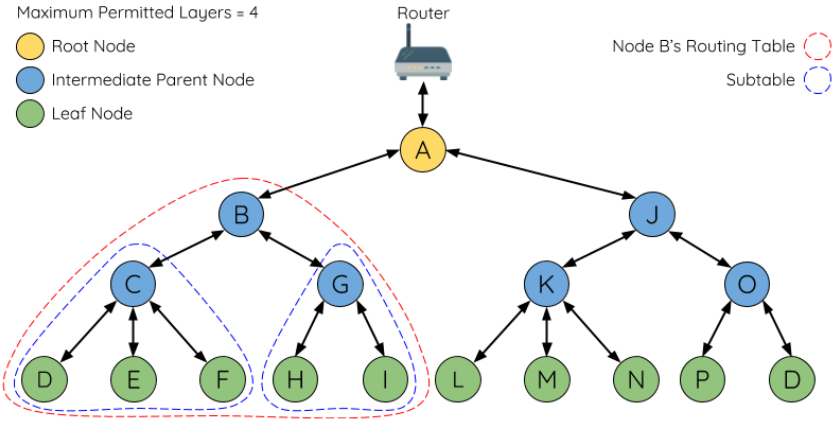


FIGURE 3.11: Preferred Parent Node Selection

Illustrated in the provided diagram, node B's routing table encompasses the MAC addresses of nodes B to I, representing the subnetwork associated with node B. Internally, node B's routing table is divided into two subtables: one comprising nodes C to F and the other containing nodes G to I, representing the subnetworks of nodes C and G, respectively.

ESP-WIFI-MESH leverages routing tables to determine the appropriate forwarding direction (upstream or downstream) for ESP-WIFI-MESH packets based on the following principles:

- If the destination MAC address of the packet falls within the current node's routing table, excluding the current node itself, the subtable containing the destination MAC address is selected, and the data packet is forwarded downstream to the child node corresponding to that subtable.
- If the destination MAC address is not found within the current node's routing table, the data packet is forwarded upstream to the parent node of the current node. Repeating this process will eventually lead the packet to reach the root node, where the routing table should encompass all nodes within the network.

3.4 Mesh network construction

Prior to commencing the establishment of an ESP-WIFI-MESH network, it is crucial to ensure that certain configuration aspects remain consistent across all nodes within the network. These aspects encompass the Mesh Network ID, router configuration, and softAP configuration, all of which must possess identical settings for each node. Please refer to the meshcfgt type for further details.

The process of constructing an ESP-WIFI-MESH network involves several steps, including the selection of a root node and the establishment of downstream connections in a layered manner until all nodes have successfully joined the network. While the precise network layout may vary due to factors like root node selection, parent node selection, and asynchronous power-on reset, the overall process can be outlined as follows:

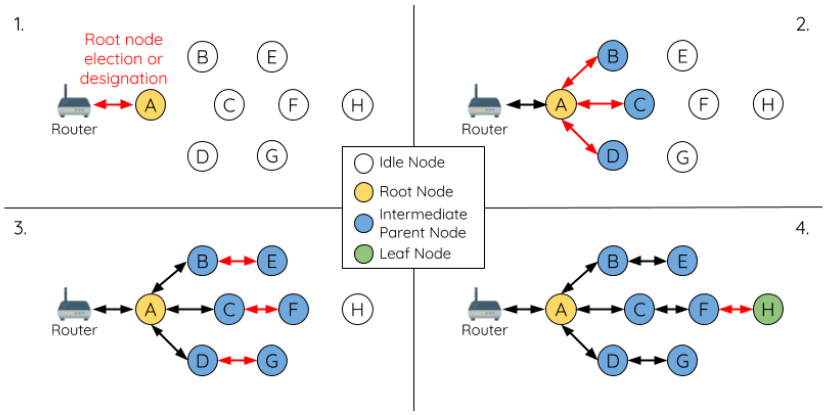


FIGURE 3.12: MESH Network Building Process

1. **Root Node Selection :** The root node can be designated during configuration (see section on User Designated Root Node), or dynamically elected based on the signal strength between each node and the router (see Automatic Root Node Selection). Once selected, the root node will connect with the router and begin allowing downstream connections to form. Referring to the figure above, node A is selected to be the root node hence node A forms an upstream connection with the router.

2. **Second Layer Formation :** Once the root node has connected to the router, idle nodes in range of the root node will begin connecting with the root node thereby forming the second layer of the network. Once connected, the second layer nodes become intermediate parent nodes (assuming maximum permitted layers > 2) hence the next layer to form. Referring to the figure above, nodes B to D are in range of the root node. Therefore nodes B to D form upstream connections with the root node and become intermediate parent nodes.

3. **Formation of remaining layers :** The remaining idle nodes will connect with intermediate parent nodes within range thereby forming a new layer in the network. Once connected, the idles nodes become intermediate parent node or leaf nodes depending on the networks maximum permitted layers. This step is repeated until there are no more idle nodes within the network or until the maximum permitted layer of the network has been reached. Referring to the figure above, nodes E/F/G connect with nodes B/C/D respectively and become intermediate parent nodes themselves.

4. **Limiting Tree Depth :** To prevent the network from exceeding the maximum permitted number of layers, nodes on the maximum layer will automatically become leaf nodes once connected. This prevents any other idle node from connecting with the leaf node thereby prevent a new layer form forming. However if an idle node has no other potential parent node, it will remain idle indefinitely. Referring to the figure above, the network's number of maximum permitted layers is set to four. Therefore when node H connects, it becomes a leaf node to prevent any downstream connections from forming.

3.4.1 Automatic Root Node Selection

The automatic selection of a root node involves an election process amongst all idle nodes based on their signal strengths with the router. Each idle node will transmit

their MAC addresses and router RSSI values via Wi-Fi beacon frames. **The MAC address is used to uniquely identify each node in the network** whilst the router RSSI is used to indicate a node's signal strength with reference to the router.

Simultaneously, each node scans for beacon frames transmitted by other idle nodes. If a node detects a beacon frame with a stronger router RSSI, it begins transmitting the contents of that beacon frame, effectively voting for the node with the stronger router RSSI. This transmission and scanning process repeats for a preconfigured minimum number of iterations (default is 10 iterations), resulting in the propagation of the beacon frame with the strongest router RSSI throughout the network.

After the completion of all iterations, each node individually evaluates its vote percentage (the number of votes divided by the number of nodes participating in the election) to determine if it should become the root node. If a node's vote percentage exceeds a preconfigured threshold (default is 90%), that node assumes the role of the root node.

The following diagram demonstrates how an ESP-WIFI-MESH network is built when the root node is automatically selected.

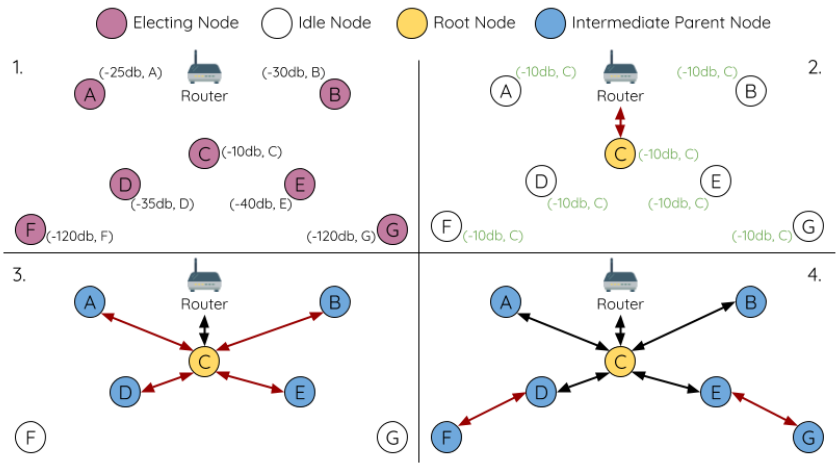


FIGURE 3.13: Root Node Election Example

1. On power-on reset, each node starts transmitting beacon frames containing their MAC addresses and router RSSIs.
2. Through multiple iterations of transmission and scanning, the beacon frame with the strongest router RSSI is propagated across the network. In the example, Node C has the strongest router RSSI (-10 dB), so its beacon frame is propagated throughout the network. All nodes participating in the election process vote for Node C, resulting in Node C obtaining a vote percentage of 100%. As a result, Node C becomes the root node and establishes a connection with the router.
3. Once Node C is connected to the router, nodes A, B, D, and E connect with Node C as it is the preferred parent node (i.e., the shallowest node). These nodes form the second layer of the network.
4. Node F and G connect with nodes D and E, respectively, and the network building process is completed.

3.4.2 User Designated Root Node:

The root node can also be designated by the user, bypassing the election process. When a root node is designated, all other nodes within the network must also forgo the election process to avoid root node conflicts. The provided diagram demonstrates the process of building an ESP-WIFI-MESH network when the root node is designated by the user.

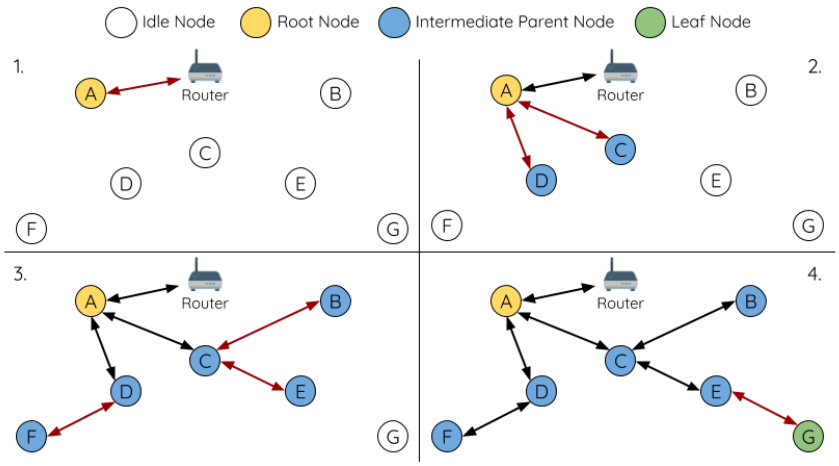


FIGURE 3.14: Root Node Designation Example (Root Node = A, Max Layers = 4)ff!bnnkl

1. Node A is designated as the root node by the user and establishes a direct connection with the router. All other nodes in the network forgo the election process.
2. Nodes C and D connect with node A as their preferred parent node, forming the second layer of the network.
3. Similarly, nodes B and E connect with node C, and node F connects with node D. Nodes B, E, and F form the third layer of the network.
4. Node G connects with node E, forming the fourth layer of the network. However, the maximum permitted number of layers in this network is set to four. Therefore, node G becomes a leaf node to prevent the formation of any new layers.

3.4.3 Parent Node Selection

Parent Node Selection in ESP-WIFI-MESH is typically self-organizing, where each node autonomously selects a preferred parent node to form an upstream connection with. This selection is based on criteria that aim to minimize the network's layers and balance downstream connections. However, users can disable self-organizing behavior to define their own selection criteria or designate specific parent nodes.

3.4.3.1 Asynchronous power-on:

reset can impact ESP-WIFI-MESH network building when nodes power on at different times. In such cases, the network structure may deviate from the ideal scenario where all nodes power on synchronously. Delayed nodes follow certain rules:

- **Rule 1:** If a root node already exists, a delayed node will not attempt to elect a new root node, even if it has a stronger signal with the router. Instead, the delayed node joins the network by connecting with a preferred parent node. If the delayed node is the designated root node, other nodes remain idle until it powers on.
- **Rule 2:** If a delayed node becomes an intermediate parent node, it may become the new preferred parent for other nodes, causing them to switch their upstream connections to the delayed node (Parent Node Switching).
- **Rule 3:** If an idle node has a designated parent node that is delayed in powering on, the idle node will not form any upstream connections until its designated parent node powers on. The idle node remains idle until then.

The effects of asynchronous power-on in network building can be illustrated through the following example.

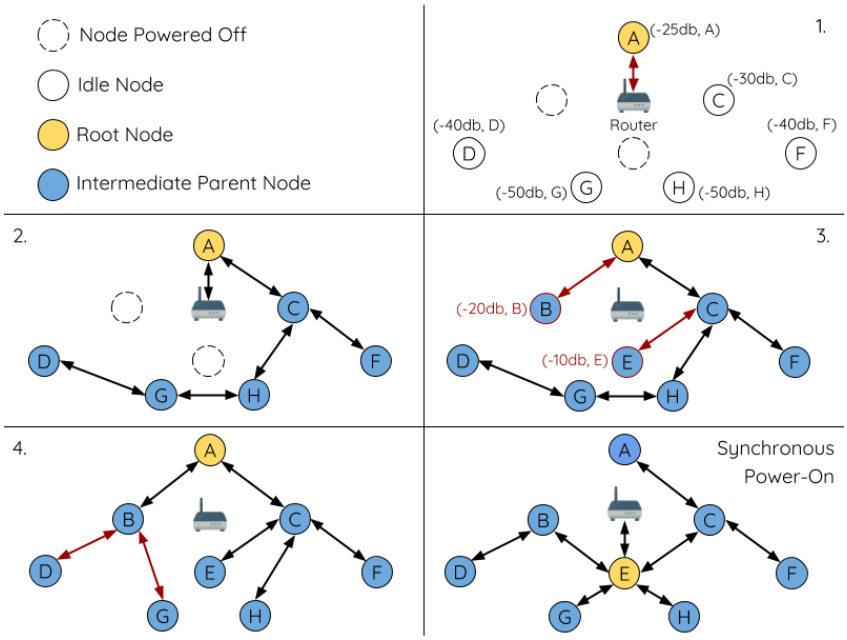


FIGURE 3.15: Network Building with Asynchronous Power On Example

1. When nodes A, C, D, F, G, and H power on simultaneously, they initiate the root node election process by broadcasting their MAC addresses and router RSSIs. Node A is elected as the root node due to having the strongest RSSI.
2. After node A becomes the root node, the other nodes start forming upstream connections with their preferred parent nodes, resulting in a network with five layers.
3. Nodes B and E experience a delay in powering on but do not attempt to become the root node, despite having stronger router RSSIs (-20 dB and -10 dB) than node A. Instead, both delayed nodes establish upstream connections with their preferred parent nodes A and C respectively. Nodes B and E become intermediate parent nodes upon connection.

4. Nodes D and G switch their upstream connections to node B, the new preferred parent node, as it is on a shallower layer (second layer node). This switch reduces the network from five layers to three layers.

3.4.3.2 Synchronous Power-On:

If all nodes powered on synchronously, •ecome the root node due to its strongest router RSSI (-10 dB). This would have resulted in a significantly different network layout compared to the network formed under asynchronous power-on conditions. However, the synchronous power-on network layout can still be achieved if the user manually switches the root node using the **esp mesh waive root** function.

3.4.3.3 Loop-back Avoidance, Detection, and Handling:

ESP-WIFI-MESH prevents loop-back, which occurs when a node forms an upstream connection with one of its descendant nodes, creating a circular connection path that breaks the tree topology. This is achieved by excluding nodes already present in the selecting node's routing table during parent selection.

3.4.3.4 Managing a Network:

ESP-WIFI-MESH is a self-healing network, capable of detecting and correcting failures in network routing. Failures can occur when a parent node with one or more child nodes breaks down or when the connection between a parent node and its child nodes becomes unstable. In such cases, child nodes autonomously select new parent nodes to maintain network interconnectivity. ESP-WIFI-MESH can handle both Root Node Failures and Intermediate Parent Node Failures.

3.4.3.5 Root Node Failure:

If the root node fails, the second layer nodes connected to it promptly detect the failure. Initially, they attempt to reconnect with the root node, but after multiple failed attempts, a new round of root node election is initialized. The second layer node with the strongest router RSSI becomes the new root node, while the remaining second layer nodes establish upstream connections with the new root node or a neighboring parent node within range.

If the root node and multiple downstream layers simultaneously fail (e.g., root node, second layer, and third layer), the shallowest functioning layer initiates the root node election. The following example illustrates a self-healing scenario after a root node failure.

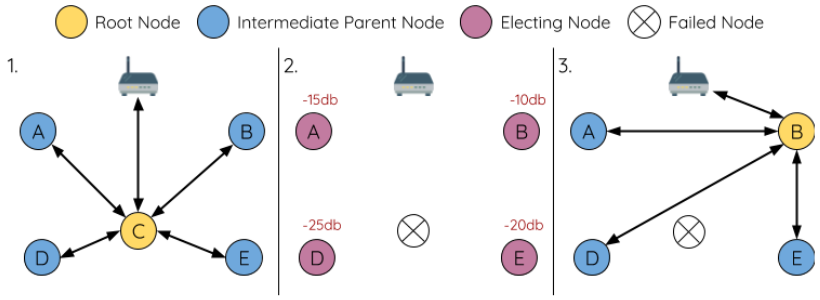


FIGURE 3.16: Self Healing From Root Node Failure

1. Node C is the root node of the network. Nodes A/B/D/E are second layer nodes connected to node C.
2. Node C breaks down. After multiple failed attempts to reconnect, the second layer nodes begin the election process by broadcasting their router RSSIs. Node B has the strongest router RSSI.
3. Node B is elected as the root node and begins accepting downstream connections. The remaining second layer nodes A/D/E form upstream connections with node B thus the network is healed and can continue operating normally.

If a designated root node breaks down, the remaining nodes will not autonomously attempt to elect a new root node as an election process will never be attempted whilst a designated root node is used.

3.4.3.6 Intermediate Parent Node Failure

If an intermediate parent node breaks down, the disconnected child nodes will initially attempt to reconnect with the parent node. After multiple failed attempts to reconnect, each child node will begin to scan for potential parent nodes.

If other potential parent nodes are available, each child node will individually select a new preferred parent node and form an upstream connection with it. If there are no other potential parent nodes for a particular child node, it will remain idle indefinitely.

The following diagram illustrates an example of self-healing from an Intermediate Parent Node breakdown.

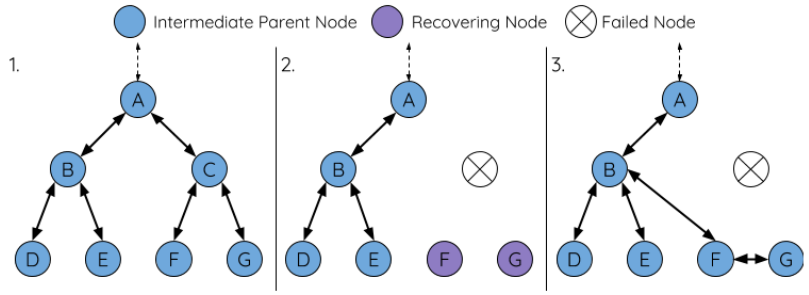


FIGURE 3.17: Self Healing From Intermediate Parent Node Failure

1. Nodes A to G form the following branch of the network.

2. If node C breaks down, nodes F and G will attempt to reconnect with node C. After multiple failed attempts, nodes F and G will select a new preferred parent node. Since node G is out of range from any other parent node, it will remain idle. Node F is in range of nodes B and E, but it will select node B as its new preferred parent node because it is shallower. After connecting with node B, node F becomes an intermediate parent node, allowing node G to connect with node F. The network is healed, but the network routing is affected and an extra layer is added.

If a child node has a designated parent node that breaks down, the child node will not attempt to connect with a new parent node; it will remain idle indefinitely.

3.4.3.7 Root Node Switching:

ESP-WIFI-MESH does not automatically switch the root node unless the root node breaks down. Even if the root node's router RSSI degrades to the point of disconnection, the root node will remain unchanged. Root node switching is the act of explicitly starting a new election such that a node with a stronger router RSSI will be elected as the new root node. This can be a useful method of adapting to degrading root node performance.

To trigger a root node switch, the current root node must explicitly call **esp mesh waive root** to initiate a new election. The current root node will signal all nodes within the network to begin transmitting and scanning for beacon frames (see Automatic Root Node Selection) whilst remaining connected to the network (i.e., not idle). If another node receives more votes than the current root node, a root node switch will be initiated. The root node will remain unchanged otherwise.

After a new root node is elected, it will send a switch request to the current root node, which will respond with an acknowledgment signifying both nodes are ready to switch. Once the acknowledgment is received, the new root node will disconnect from its parent and promptly form an upstream connection with the router, thereby becoming the new root node of the network. The previous root node will disconnect from the router while maintaining all of its downstream connections and enter the idle state. The previous root node will then begin scanning for potential parent nodes and selecting a preferred parent.

The following diagram illustrates an example of a root node switch:

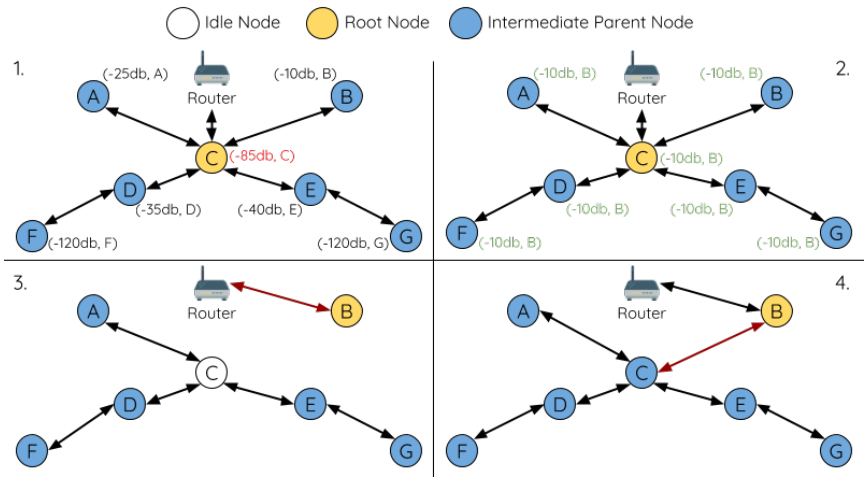


FIGURE 3.18: root node switch

3.4.3.8 Root Node Switch Example

1. Node C, the current root node, experiences degraded signal strength (-85 dB) with the router. As a result, node C triggers a new election process, and all nodes start transmitting and scanning for beacon frames while maintaining their connections.
2. After multiple rounds of transmission and scanning, node B is elected as the new root node. Node B sends a switch request to node C, which responds with an acknowledgment.
3. Node B disconnects from its previous parent and establishes a connection with the router, becoming the new root node of the network. Node C disconnects from the router, enters the idle state, and begins scanning for and selecting a new preferred parent node. Throughout this process, node C maintains all its downstream connections.
4. Node C selects node B as its preferred parent node, establishes an upstream connection, and becomes a second-layer node. The network layout remains similar after the switch, as node C still maintains the same subnetwork. However, due to the switch, each node in node C's subnetwork is placed one layer deeper. Parent Node Switching may further adjust the network layout if any nodes have a new preferred parent node as a result of the root node switch.

- Root node switching requires an election and is only supported in self-organized ESP-WIFI-MESH networks. It cannot occur if a designated root node is used.

3.4.3.9 Parent Node Switching

Parent Node Switching refers to a child node switching its upstream connection to a parent node located on a shallower layer. This switching occurs autonomously, meaning that a child node automatically changes its upstream connection if a potential parent node from a shallower layer becomes available (e.g., due to an Asynchronous Power-on Reset).

All potential parent nodes periodically transmit beacon frames, allowing a child node to scan for the availability of shallower parent nodes. Through parent node

switching, a self-organized ESP-WIFI-MESH network can dynamically adjust its network layout to ensure optimal RSSI for each connection and minimize the number of layers in the network.

3.4.3.10 Data Transmission

ESP-WIFI-MESH networks use ESP-WIFI-MESH packets for data transmission. These packets are encapsulated within the frame body of a Wi-Fi data frame. In a multi-hop data transmission within an ESP-WIFI-MESH network, each wireless hop involves a different Wi-Fi data frame carrying a single ESP-WIFI-MESH packet.

The diagram below illustrates the structure of an ESP-WIFI-MESH packet and its relationship with a Wi-Fi data frame.

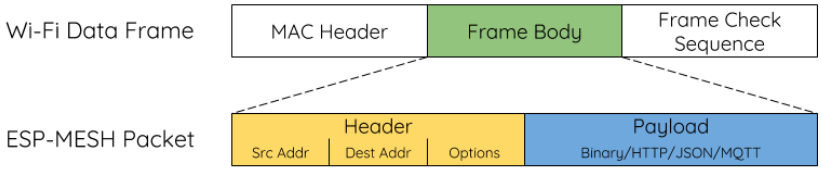


FIGURE 3.19: ESP-WIFI-MESH Packet

An ESP-WIFI-MESH packet consists of a header and a payload. The header contains the MAC addresses of the source and destination nodes, while the options field holds information about special packet types, such as group transmissions or packets from the external IP network. The macros MESH-OPT-SEND-GROUP and MESH-OPT-RECV-DS-ADDR are used to handle these options.

The payload of an ESP-WIFI-MESH packet carries the actual application data, which can be in raw binary format or encoded using protocols like HTTP, MQTT, and JSON. The mesh-proto-t type specifies the application layer protocol.

When sending a packet to the external IP network, the destination address field in the header contains the IP address and port of the target server instead of a MAC address. The root node takes care of forming the outgoing TCP/IP packet.

3.4.3.11 Broadcasting

Broadcasting allows a packet to be transmitted simultaneously to all nodes within the network. Specific methods are implemented to optimize broadcasting:

1. Intermediate parent node receiving a broadcast: The packet is forwarded to each child node.
2. Intermediate parent node as the source node of the broadcast: The packet is transmitted upstream to the parent node and downstream to each child node.
3. Intermediate parent node receiving a broadcast from its child node: The packet is forwarded to the parent node and the remaining child nodes.
4. Leaf node as the source node of the broadcast: The packet is transmitted directly to the parent node.
5. Root node as the source node of the broadcast: The packet is sent to all child nodes.

6. Root node receiving a broadcast from one of its child nodes: The packet is forwarded to the remaining child nodes.
7. Node discarding a broadcast packet: If a node receives a broadcast packet with a source address matching its own MAC address, it discards the packet.

3.4.3.12 Upstream Flow Control

Parent nodes control the upstream data flow of their child nodes by allocating a receiving window quota. A child node applies for a receiving window before transmitting upstream. The size of the receiving window is dynamically adjusted based on a comparison of sequence numbers. If the receiving window is depleted, the child node requests another window before continuing transmission.

Note that ESP-WIFI-MESH does not support downstream flow control, and packet loss may occur during upstream transmissions due to Parent Node Switching.

To ensure proper data flow, downstream nodes need to be aware of the root node's connection status with the external IP network. ESP-WIFI-MESH addresses this issue by allowing the root node to broadcast its external IP network connection status to all other nodes using the esp-mesh-post-toDS-state function.

3.4.3.13 Bi-Directional Data Stream

A diagram illustrating the various network layers involved in an ESP-WIFI-MESH Bidirectional Data Stream is available, but it cannot be displayed here.

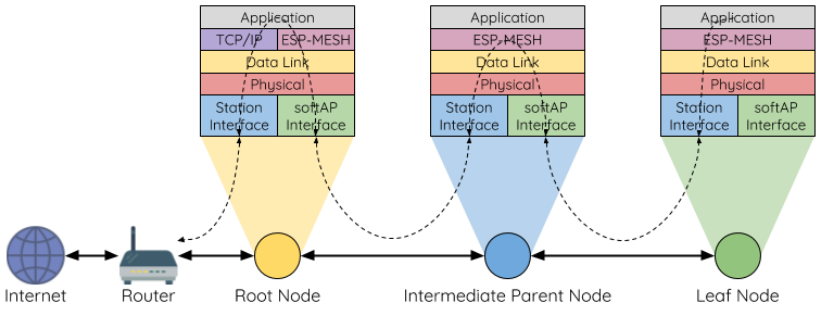


FIGURE 3.20: Bidirectional Data Stream

In traditional Wi-Fi networks, it's important for the access point (AP) and connected stations to operate on the same channel to avoid interference. To adapt to changing conditions, Wi-Fi networks implement network channel switching mechanisms. In an infrastructure Wi-Fi network, the AP triggers network channel switches by including a Channel Switch Announcement (CSA) element in beacon frames. This element informs connected stations about an upcoming channel switch.

3.4.3.14 Network Channel Switching

ESP-WIFI-MESH also utilizes beacon frames with CSA elements for network channel switching. However, in a multi-hop network like ESP-WIFI-MESH, the switching process is more complex as beacon frames may not reach all nodes in a single hop. Therefore, the network relies on nodes to forward the CSA element to ensure propagation.

When an intermediate parent node receives a beacon frame with a CSA element, it includes the CSA element in its own transmitted beacon frame, keeping the New Channel Number and Channel Switch Count the same. This ensures that all nodes in the network receive the CSA element and can synchronize their channel switches using the Channel Switch Count, although with a slight delay due to forwarding.

3.5 Network Map

Generating a Wireless Sensor Network (WSN) topology using the PainlessMesh library can be a complex yet rewarding task. This paragraph will provide a comprehensive overview of the steps involved in generating a WSN topology using the PainlessMesh library.

The first step in generating a WSN topology is to understand the basic concepts of the PainlessMesh library. PainlessMesh is an Arduino-based library that allows easy creation of mesh networks using the ESP8266 or ESP32 boards. It provides a simple and efficient way to establish communication between multiple nodes in a network, forming a self-healing and self-configuring mesh topology.

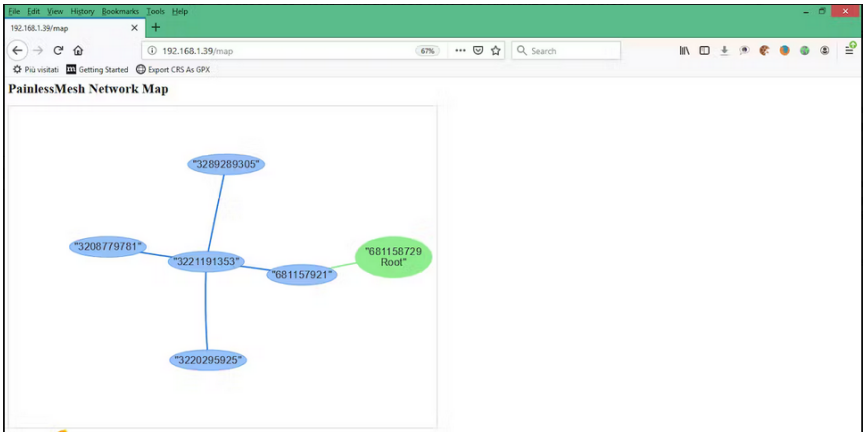


FIGURE 3.21: Bidirectional Data Stream

Once the web server is set up, the next step is to establish communication between the PainlessMesh network and the web server. The PainlessMesh library provides functions to send and receive data between nodes in the network. By leveraging these functions, the nodes can periodically send their network topology information, such as node IDs, positions, and connectivity details, to the web server.

On the web server side, a WebSocket connection can be established to enable real-time communication between the PainlessMesh network and the browser. The WebSocket protocol allows for bidirectional communication, ensuring that data can be transmitted from the server to the browser and vice versa. The server-side code should handle incoming data from the PainlessMesh network and broadcast it to the connected browser clients.

On the browser side, JavaScript can be used to establish a WebSocket connection with the server. Once the connection is established, the browser can receive updates from the PainlessMesh network in real-time. The received data can then be processed and used to visualize the WSN topology.

3.6 PainlessMesh library

painlessMesh is a library that takes care of the particulars of creating a simple mesh network using esp8266 and esp32 hardware. The goal is to allow the programmer to work with a mesh network without having to worry about how the network is structured or managed.

3.6.1 Basic node code

```
#include "painlessMesh.h"

#define   MESH_PREFIX     "whateverYouLike"
#define   MESH_PASSWORD   "somethingSneaky"
#define   MESH_PORT       5555

Scheduler userScheduler; // to control your personal task
painlessMesh  mesh;

// User stub
void sendMessage() ; // Prototype so PlatformIO doesn't complain

Task taskSendMessage( TASK_SECOND * 1 , TASK_FOREVER, &sendMessage );

void sendMessage() {
  String msg = "Hello from node ";
  msg += mesh.getNodeId();
  mesh.sendBroadcast( msg );
  taskSendMessage.setInterval( random( TASK_SECOND * 1, TASK_SECOND * 5
     ));
}

// Needed for painless library
void receivedCallback( uint32_t from, String &msg ) {
  Serial.printf("startHere: Received from %u msg=%s\n", from, msg.c_str
    ());
}

void newConnectionCallback(uint32_t nodeId) {
    Serial.printf("--> startHere: New Connection, nodeId = %u\n",
    nodeId);
}

void changedConnectionCallback() {
  Serial.printf("Changed connections\n");
}

void nodeTimeAdjustedCallback(int32_t offset) {
    Serial.printf("Adjusted time %u. Offset = %d\n", mesh.getNodeTime()
    ,offset);
}
```

```
void setup() {
  Serial.begin(115200);

//mesh.setDebugMsgTypes( ERROR | MESH_STATUS | CONNECTION | SYNC |
    COMMUNICATION | GENERAL | MSG_TYPES | REMOTE ); // all types on
  mesh.setDebugMsgTypes( ERROR | STARTUP );  // set before init() so
    that you can see startup messages

  mesh.init( MESH_PREFIX, MESH_PASSWORD, &userScheduler, MESH_PORT );
  mesh.onReceive(&receivedCallback);
  mesh.onNewConnection(&newConnectionCallback);
  mesh.onChangedConnections(&changedConnectionCallback);
  mesh.onNodeTimeAdjusted(&nodeTimeAdjustedCallback);

  userScheduler.addTask( taskSendMessage );
  taskSendMessage.enable();
}

void loop() {
  // it will run the user scheduler as well
  mesh.update();
}
```

3.6.2 MQTT Bridge node

This is for the root node that connects to the raspberry pi.

```
#include <Arduino.h>
#include <painlessMesh.h>
#include <PubSubClient.h>
#include <WiFiClient.h>

#define   MESH_PREFIX     "whateverYouLike"
#define   MESH_PASSWORD   "somethingSneaky"
#define   MESH_PORT       5555

#define   STATION_SSID     "YourAP_SSID"
#define   STATION_PASSWORD "YourAP_PWD"

#define HOSTNAME "MQTT_Bridge"

// Prototypes
void receivedCallback( const uint32_t &from, const String &msg );
void mqttCallback(char* topic, byte* payload, unsigned int length);

IPAddress getlocalIP();

IPAddress myIP(0,0,0,0);
IPAddress mqttBroker(192, 168, 1, 1);
```

```
painlessMesh  mesh;
WiFiClient wifiClient;
PubSubClient mqttClient(mqttBroker, 1883, mqttCallback, wifiClient);

void setup() {
  Serial.begin(115200);

  mesh.setDebugMsgTypes( ERROR | STARTUP | CONNECTION );  // set before
     init() so that you can see startup messages

  // Channel set to 6. Make sure to use the same channel for your mesh
    and for you other
  // network (STATION_SSID)
  mesh.init( MESH_PREFIX, MESH_PASSWORD, MESH_PORT, WIFI_AP_STA, 6 );
  mesh.onReceive(&receivedCallback);

  mesh.stationManual(STATION_SSID, STATION_PASSWORD);
  mesh.setHostname(HOSTNAME);

  // Bridge node, should (in most cases) be a root node. See [the wiki
    ](https://gitlab.com/painlessMesh/painlessMesh/wikis/Possible-
    challenges-in-mesh-formation) for some background
  mesh.setRoot(true);
  // This node and all other nodes should ideally know the mesh
    contains a root, so call this on all nodes
  mesh.setContainsRoot(true);
}

void loop() {
  mesh.update();
  mqttClient.loop();

  if(myIP != getlocalIP()){
    myIP = getlocalIP();
    Serial.println("My␣IP␣is␣" + myIP.toString());

    if (mqttClient.connect("painlessMeshClient")) {
      mqttClient.publish("painlessMesh/from/gateway","Ready!");
      mqttClient.subscribe("painlessMesh/to/#");
    }
  }
}

void receivedCallback( const uint32_t &from, const String &msg ) {
  Serial.printf("bridge:␣Received␣from␣%u␣msg=%s\n", from, msg.c_str())
    ;
  String topic = "painlessMesh/from/" + String(from);
  mqttClient.publish(topic.c_str(), msg.c_str());
}
```

```
void mqttCallback(char* topic, uint8_t* payload, unsigned int length) {
  char* cleanPayload = (char*)malloc(length+1);
  memcpy(cleanPayload, payload, length);
  cleanPayload[length] = '\0';
  String msg = String(cleanPayload);
  free(cleanPayload);

  String targetStr = String(topic).substring(16);

  if(targetStr == "gateway")
  {
    if(msg == "getNodes")
    {
      auto nodes = mesh.getNodeList(true);
      String str;
      for (auto &&id : nodes)
        str += String(id) + String(" ");
      mqttClient.publish("painlessMesh/from/gateway", str.c_str());
    }
  }
  else if(targetStr == "broadcast")
  {
    mesh.sendBroadcast(msg);
  }
  else
  {
    uint32_t target = strtoul(targetStr.c_str(), NULL, 10);
    if(mesh.isConnected(target))
    {
      mesh.sendSingle(target, msg);
    }
    else
    {
      mqttClient.publish("painlessMesh/from/gateway", "Client not
    connected!");
    }
  }
}

IPAddress getlocalIP() {
  return IPAddress(mesh.getStationIP());
}
```

3.7 Task sheduling in esp32

Task scheduling in embedded systems is an important aspect of real-time systems, where tasks must be executed in a specific order and within certain time constraints.

This is especially critical in safety-critical applications, such as aerospace and automotive systems, where any delays or missed deadlines can have serious consequences.

There are various approaches to task scheduling in embedded systems, including static scheduling and dynamic scheduling. In static scheduling, the task execution order and timing are determined at design-time and do not change during runtime. This approach is useful for applications with fixed, predictable task sequences. Dynamic scheduling, on the other hand, allows for runtime task reordering and adaptation to changing system conditions.

One common technique for task scheduling in embedded systems is the use of priority-based scheduling algorithms. In this approach, each task is assigned a priority level based on its relative importance and deadline requirements. The scheduler then selects the highest priority task to execute next, ensuring that the most critical tasks are completed first.

Another approach is to use time-based scheduling algorithms, where tasks are scheduled based on their execution time requirements. For example, tasks with shorter execution times may be prioritized over longer tasks to minimize overall system latency.

There are also several scheduling policies that can be used to determine task execution order, including earliest deadline first (EDF), rate monotonic (RM), and deadline monotonic (DM). EDF scheduling prioritizes tasks based on their earliest deadline, while RM and DM prioritize tasks based on their period and deadline, respectively.

3.7.1 Task Scheduler library

A lightweight implementation of cooperative multitasking (task scheduling). An easier alternative to preemptive programming and frameworks like FreeRTOS.

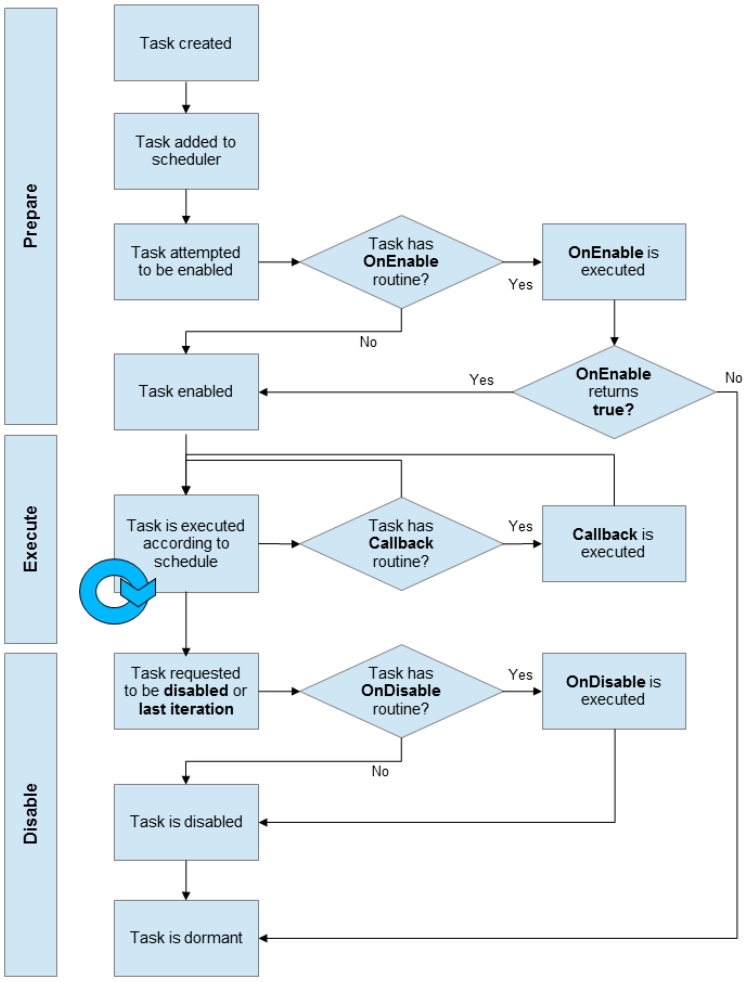


FIGURE 3.22: Task Scheduler

3.8 XIAO ESP32C3 sensor chip

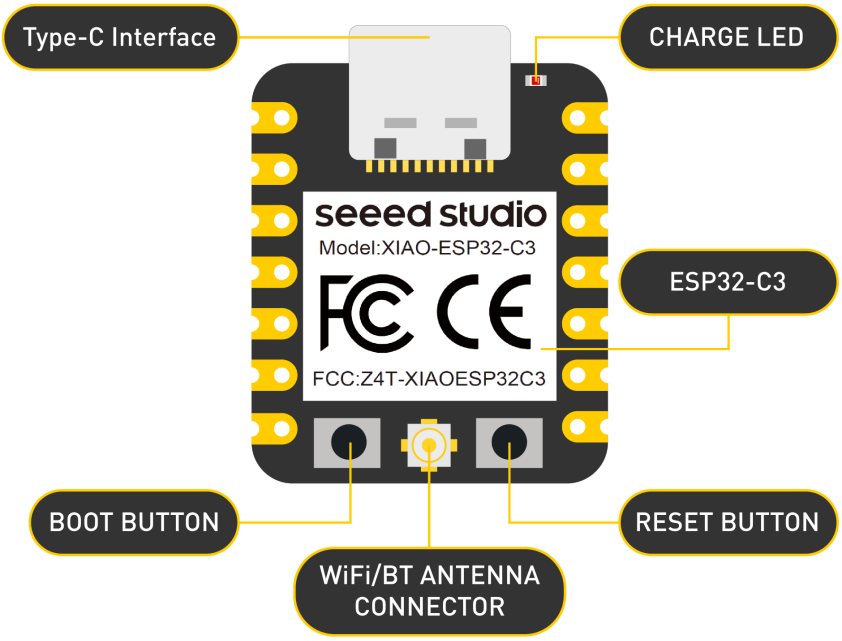


FIGURE 3.23: Component overview 1

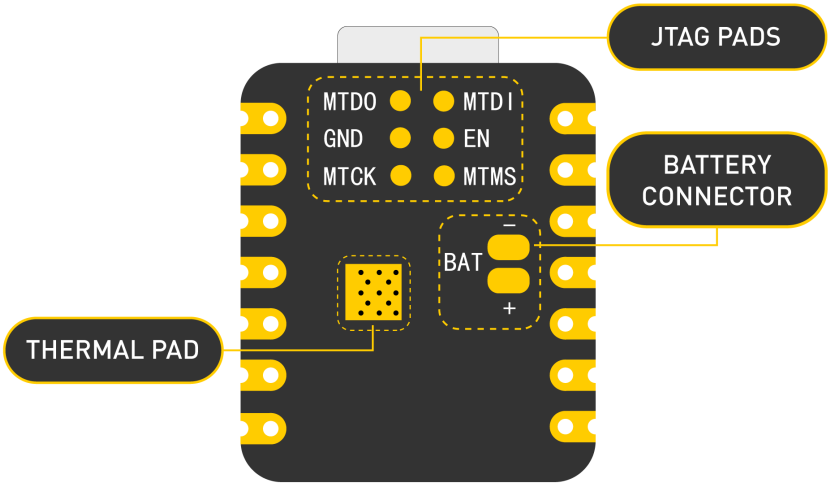


FIGURE 3.24: Component overview 2

Seeed Studio XIAO ESP32C3 is an IoT mini development board based on the Espressif ESP32-C3 WiFi/Bluetooth dual-mode chip. ESP32-C3 is a 32-bit RISC-V CPU, which includes an FPU (Floating Point Unit) for 32-bit single-precision arithmetic with powerful computing power. It has excellent radio frequency performance, supporting IEEE 802.11 b/g/n WiFi, and Bluetooth 5 (LE) protocols. This board comes included with an external antenna to increase the signal strength for your wireless applications. It also has a small and exquisite form-factor combined with a single-sided surface-mountable design. It is equipped with rich interfaces and has 11 digital I/O that can be used as PWM pins and 4 analog I/O that can be used as ADC pins. It supports four serial interfaces such as UART, I2C and SPI. There is also a small reset button and a bootloader mode button on the board. XIAO ESP32C3 is fully compatible with the Grove Shield for Seeeduino XIAO and Seeeduino XIAO Expansion board except for the Seeeduino XIAO Expansion board, the SWD spring

contacts on the board will not be compatible.

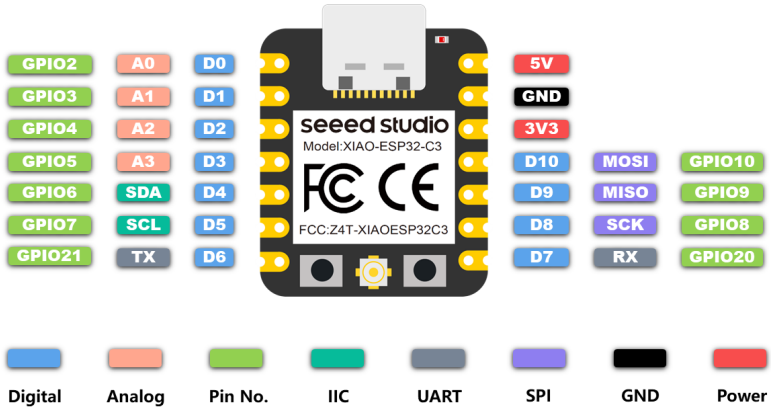


FIGURE 3.25: XIAO ESP32C3 Pinout diagram

With regard to the features highlighted above, XIAO ESP32C3 is positioned as a high-performance, low-power, cost-effective IoT mini development board, suitable for low-power IoT applications and wireless wearable applications.

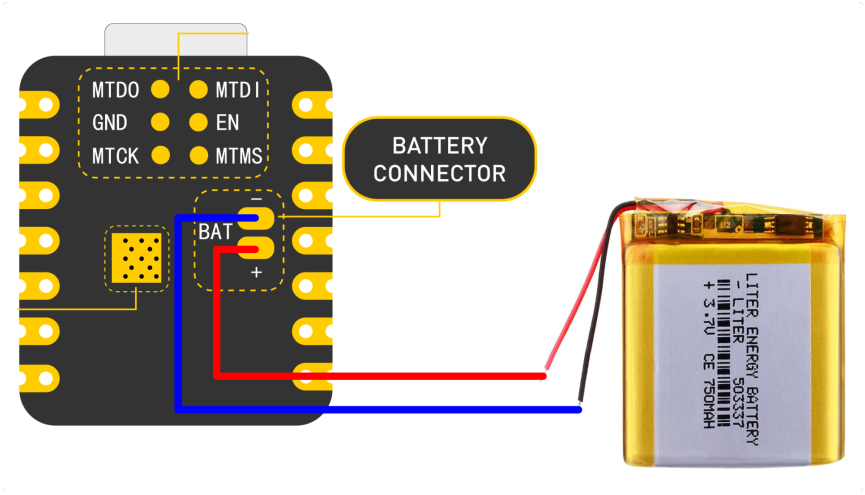


FIGURE 3.26: XIAO ESP32C3 battery connection

The XIAO ESP32C3 is capable of using a 3.7V lithium battery as the power supply input. You can refer to the following diagram for the wiring method.

3.9 24GHz mmWave Sensor

24GHz mmWave Sensor - Human Static Presence Module Lite is an antenna-integrated, health-friendly mmwave radar sensor that applies FMCW ranging technology, with operation in 24GHz, for implementation of human presence, independently of environmental influences. This is also a personalized radar that users can configure underlying parameters of it to determine detected functions.

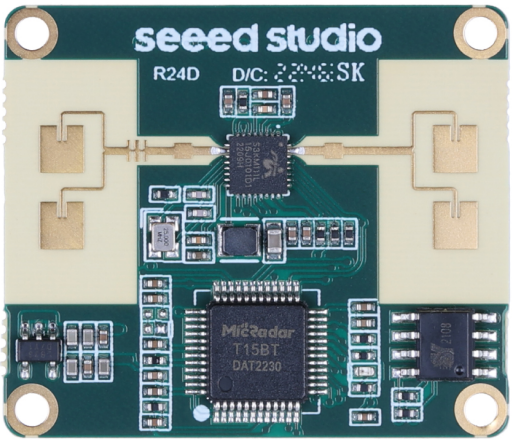


FIGURE 3.27: 24GHz mmWave Sensor

This is the pinout diagram for the sensor.

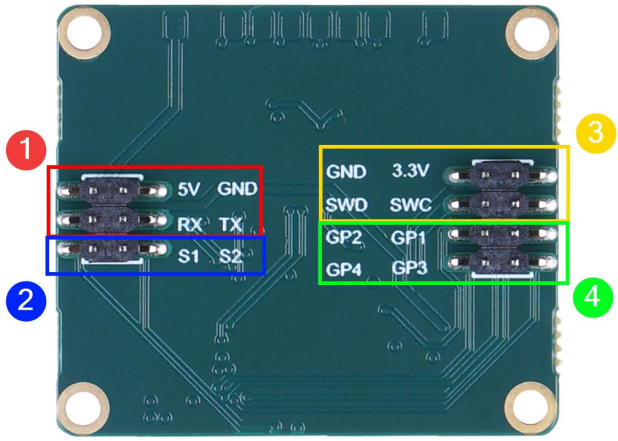


FIGURE 3.28: 24GHz mmWave Sensor connextions

here we have the connections to the XIAO ESP32C3.

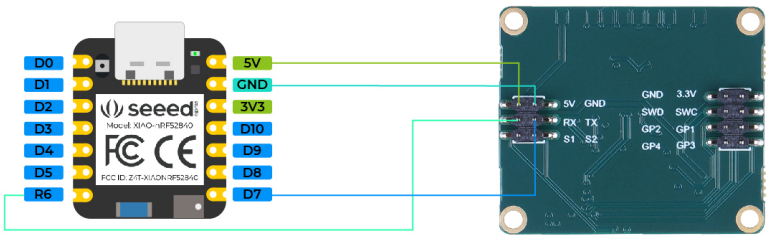


FIGURE 3.29: 24GHz mmWave Sensor

The code for movment detection nodes, we used the following.

```
#include "Arduino.h"
#include <humanstaticLite.h>

//#include <SoftwareSerial.h>
// Choose any two pins that can be used with SoftwareSerial to RX & TX
//#define RX_Pin A2
//#define TX_Pin A3

//SoftwareSerial mySerial = SoftwareSerial(RX_Pin, TX_Pin);

// we'll be using software serial
//HumanStaticLite radar = HumanStaticLite(&mySerial);

// can also try hardware serial with
HumanStaticLite radar = HumanStaticLite(&Serial1);

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  Serial1.begin(115200);

  //  mySerial.begin(115200);

  while(!Serial);   //When the serial port is opened, the program
    starts to execute.

  Serial.println("Ready");
}

void loop() {
  // put your main code here, to run repeatedly:
  radar.HumanStatic_func(true);    //Turn on printing of human movement
     sign parameters
  if(radar.radarStatus != 0x00){
    switch(radar.radarStatus){
      Serial.println(radar.radarStatus);
      case SOMEONE:
```

```
        Serial.println("Someone␣is␣here.");
        Serial.println("--------------------------------");
        break;
      case NOONE:
        Serial.println("Nobody␣here.");
        Serial.println("--------------------------------");
        break;
      case NOTHING:
        Serial.println("No␣human␣activity␣messages.");
        Serial.println("--------------------------------");
        break;
      case SOMEONE_STOP:
        Serial.println("Someone␣stop");
        Serial.println("--------------------------------");
        break;
      case SOMEONE_MOVE:
        Serial.println("Someone␣moving");
        Serial.println("--------------------------------");
        break;
      case HUMANPARA:
        Serial.print("The␣parameters␣of␣human␣body␣signs␣are:␣");
        Serial.println(radar.bodysign_val, DEC);
        Serial.println("--------------------------------");
        break;
      case SOMEONE_CLOSE:
        Serial.println("Someone␣is␣closing");
        Serial.println("--------------------------------");
        break;
      case SOMEONE_AWAY:
        Serial.println("Someone␣is␣staying␣away");
        Serial.println("--------------------------------");
        break;
      case DETAILMESSAGE:
        Serial.print("Spatial␣static␣values:␣");
        Serial.println(radar.static_val);
        Serial.print("Distance␣to␣stationary␣object:␣");
        Serial.print(radar.dis_static);
        Serial.println("␣m");

        Serial.print("Spatial␣dynamic␣values:␣");
        Serial.println(radar.dynamic_val);

        Serial.print("Distance␣from␣the␣movement␣object:␣");
        Serial.print(radar.dis_move);
        Serial.println("␣m");

        Serial.print("Speed␣of␣moving␣object:␣");
        Serial.print(radar.speed);
        Serial.println("␣m/s");
        Serial.println("--------------------------------");
        break;
```

```
    }
  }
  delay(200);
}
```

3.10 PIR sensor

The PIR motion sensor is ideal to detect movement. PIR stand for "Passive Infrared". Basically, the PIR motion sensor measures infrared light from objects in its field of view.

So, it can detect motion based on changes in infrared light in the environment. It is ideal to detect if a human has moved in or out of the sensor range.

The sensor in the figure above has two built-in potentiometers to adjust the delay time (the potentiometer at the left) and the sensitivity (the potentiometer at the right).

```
int led = 13;                // the pin that the LED is atteched to
int sensor = 2;              // the pin that the sensor is atteched to
int state = LOW;             // by default, no motion detected
int val = 0;                 // variable to store the sensor status (
    value)

void setup() {
  pinMode(led, OUTPUT);      // initalize LED as an output
  pinMode(sensor, INPUT);    // initialize sensor as an input
  Serial.begin(9600);        // initialize serial
}

void loop(){
  val = digitalRead(sensor);   // read sensor value
  if (val == HIGH) {           // check if the sensor is HIGH
    digitalWrite(led, HIGH);   // turn LED ON
    delay(100);                // delay 100 milliseconds

    if (state == LOW) {
      Serial.println("Motion detected!");
      state = HIGH;       // update variable state to HIGH
    }
  }
  else {
      digitalWrite(led, LOW); // turn LED OFF
      delay(200);             // delay 200 milliseconds

      if (state == HIGH){
        Serial.println("Motion stopped!");
        state = LOW;       // update variable state to LOW
    }
  }
```

```
}
```

3.11 DHT11 nodes (Tempreture and humidity sensors)

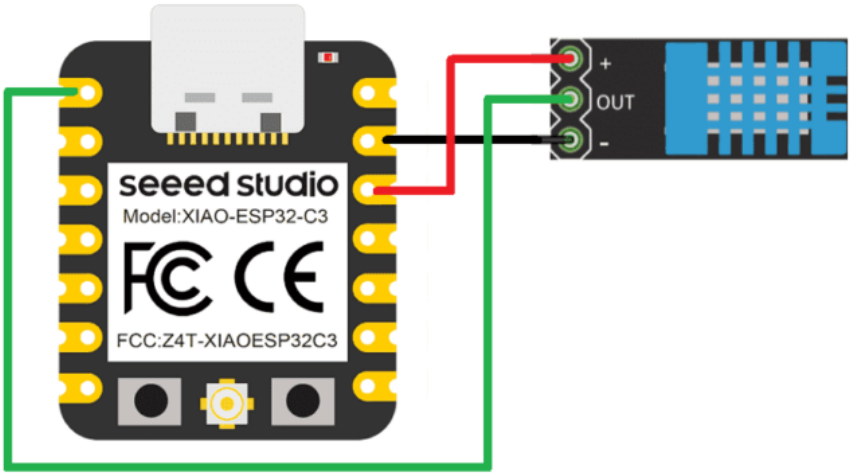


FIGURE 3.30: DHT11 sensor

```
#include "DHT.h"
#define DHTPIN 5     // Digital pin connected to the DHT sensor

#define DHTTYPE DHT11   // DHT 11
DHT dht(DHTPIN, DHTTYPE);

void setup()
{
  Serial.begin(9600);
  Serial.println(F("DHTxx␣test!"));
  dht.begin();
  delay(2000);
}

void loop()
{
  float h = dht.readHumidity();
  float t = dht.readTemperature();

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t))
  {
    Serial.println(F("Failed␣to␣read␣from␣DHT␣sensor!"));
    return;
  }
  Serial.print(F("Humidity:␣"));
  Serial.print(h);
  Serial.println("%");
```

```
  Serial.print(F("Temperature: "));
  Serial.print(t);
  Serial.println(F("C "));
  Serial.println("");
  delay(2000);
}
```

Chapter 4

Pateneted Prototype for "Border Surveillance and Monitoring"

This invention "Border surveillance and monitoring using solar powered Wireless Sensor Network equipped with 24GHz mmWave sensors and GPS Modules for tracking and ESP-NOW and LoRa protocols for communication" is a technical solution that involves the use of a wireless sensor network to monitor and track all activities at border areas or any large scale spaces. The system can detect and track illegal border crossings, suspicious activities, and other potential security threats. The data collected can be used to improve border security and provide real-time information to law enforcement agencies.

Our border surveillance system utilizes wireless sensor networks (WSN) built with ESP32 chips, which are powered by lithium batteries charged with solar power. The system employs advanced technologies such as ESP-NOW and LoRa for short and long-distance communication, respectively. The nodes of the system also include GPS chips for location tracking and 24GHz microwave sensors for detecting movements. The sensor nodes communicate with a base station made with a Raspberry Pi via the MQTT protocol, and data is transmitted to a central base station for monitoring and control. Additionally, the system allows for the tracking of the sensor nodes' locations via mobile phones.

Overall, this border surveillance system provides an efficient and cost-effective solution for monitoring and securing border areas. By utilizing advanced technologies such as wireless sensor networks, GPS tracking, and microwave sensors, the system can detect any unauthorized movement in real-time, allowing for swift action to be taken. Moreover, the use of solar power and lithium batteries makes the system environmentally friendly and cost-effective, reducing the need for frequent maintenance and electricity supply.

4.1 Objectives

1. **Border security:** The primary objective of this invention is to enhance border security by providing a reliable and efficient system for monitoring and detecting any unauthorized activity across the border. The sensors can detect and report any movement of people or vehicles in real-time, which can help authorities to take appropriate action.
2. **Cost-effectiveness:** The use of low-cost and low-power wireless sensors, solar power, and LoRa communication technology makes this system cost-effective compared to other border surveillance systems that require a constant supply of electricity and high-speed internet connectivity.
3. **Real-time monitoring:** The use of GPS modules and real-time data transmission using LoRa for long-range and ESP-NOW for shorter-range communication allows authorities to monitor the movement of people and vehicles across the border in real time. This can help in preventing illegal activities such as smuggling and trafficking.

4.2 How the Invention is made

A wireless sensor network is a collection of small, low-power, and low-cost wireless devices called sensors that are distributed over a wide area to monitor various environmental and physical conditions. The sensors in a WSN are typically equipped with various types of sensors to measure different parameters.

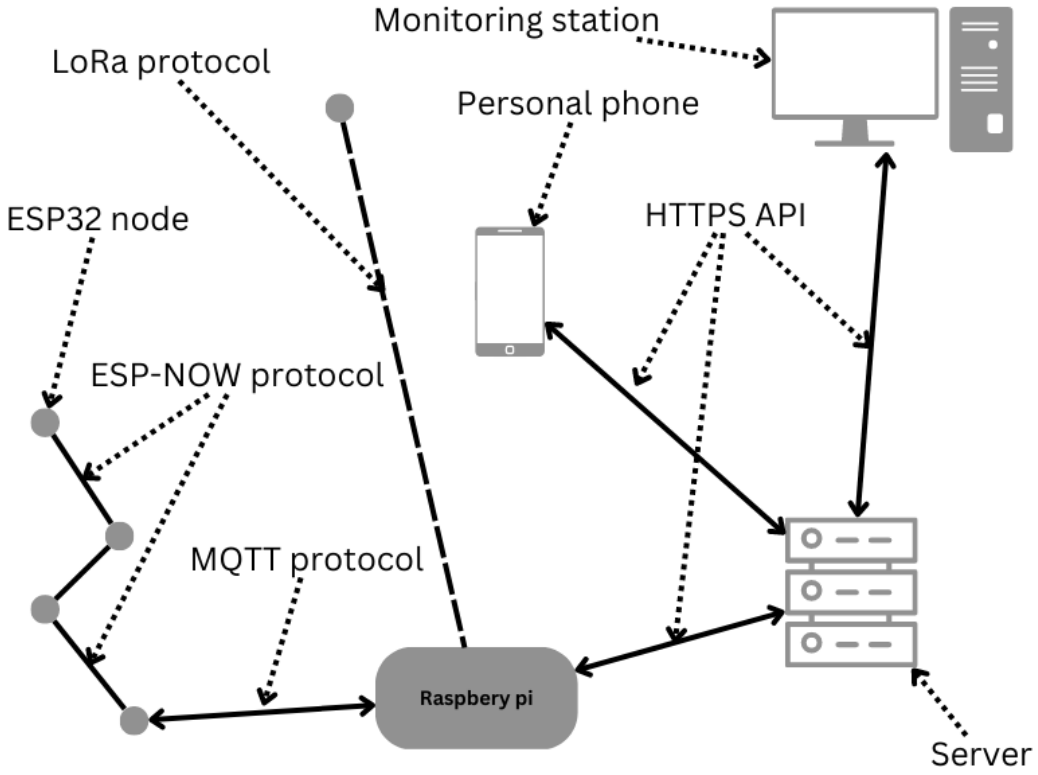


FIGURE 4.1: Sensor functioning components

n the case of the described invention, the wireless sensor nodes are equipped with ESP32 modules, which are small, low-power, and low-cost microcontrollers. The nodes are also equipped with 24GHz mmWave sensors, which are used for short-range radar sensing applications. These sensors operate at frequencies above 30 GHz and provide high resolution and accuracy in detecting targets.

The system utilizes 24GHz mmWave sensors to detect and track movement, along with GPS modules to provide location data. The mmWave sensors use a high-frequency electromagnetic signal to detect objects and measure their distance and speed, making them suitable for detecting human and vehicle activity at long distances.

The wireless sensor network is powered by solar panels, which ensures that the system can operate independently of the electrical grid and in remote areas. The sensors are interconnected and communicate with each other wirelessly, sending data to a central server where it is analyzed and processed.

Finally, ESP-NOW is a low-power, low-latency, and high-throughput protocol developed by Espressif Systems that operates over the 2.4GHz frequency band. By incorporating ESP-NOW into the system, the sensors can communicate with each other over short distances without the need for a Wi-Fi or cellular network, making it an ideal solution for border areas where such networks may be unavailable or unreliable. ESP-NOW also offers the advantage of being highly secure and resistant to interference, which is important in border security applications where unauthorized access to the network or data can pose a significant risk.

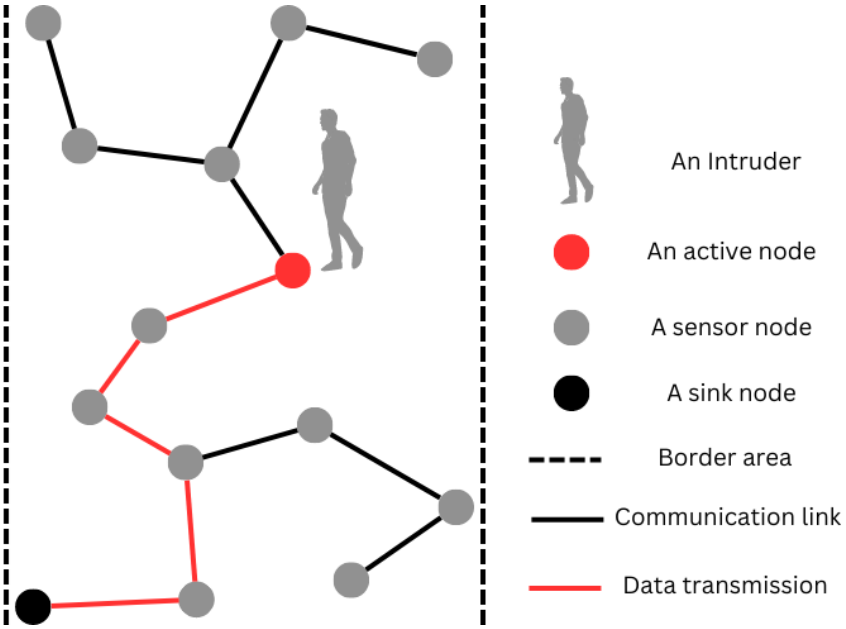


FIGURE 4.2: Sensor node components

The wireless sensors are also equipped with LoRa (Long Range) communication technology, which is a low-power, long-range wireless communication protocol that allows the sensors to transmit data over long distances without consuming too much power. This is important as it acts as a backup for ESP-NOW protocol and to increases the life of the system by allowing the sensor nodes to communicate with each other over long distances, as sensor nodes start to die over time, it will ensure the system will still be functional as it is able to send data over longer distances.

Incorporating the MQTT protocol can further enhance the functionality of the system. MQTT is a lightweight and efficient messaging protocol designed for use in machine-to-machine (M2M) and Internet of Things (IoT) applications. In this case, the closest node to the Raspberry Pi can use the MQTT protocol to communicate with it. This allows for the transmission of real-time data from the sensors to the Raspberry Pi. Overall, the described invention is a sophisticated system that combines various advanced technologies to provide an effective and reliable border surveillance and monitoring solution.

4.3 How the Invention works

Border surveillance and monitoring is a critical aspect of national security, and technology has played an essential role in enhancing this capability. A recent development in this area is the use of wireless sensor networks (WSN) equipped with 24GHz mmWave sensors and GPS modules for border surveillance and monitoring. This system is designed to provide reliable and real-time data on human and vehicle activity along the border, enabling quick and appropriate action when necessary.

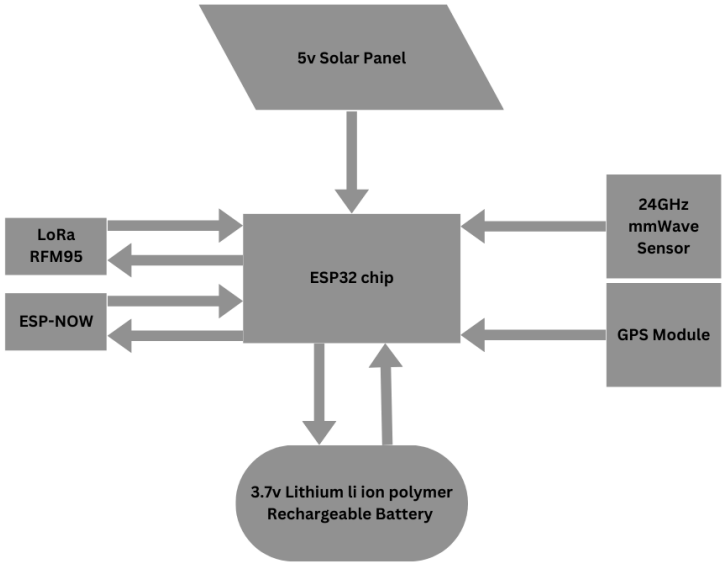
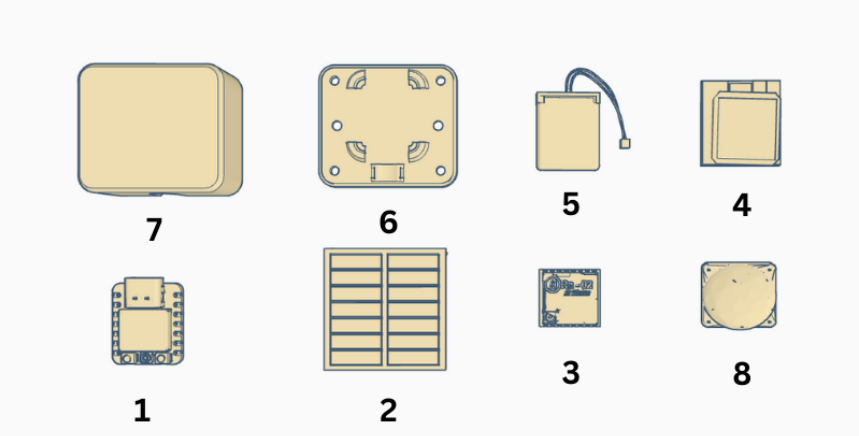


FIGURE 4.3: Example of Intruder detection

The system works by creating a wireless sensor network in the area to be monitored. The nodes are scattered across the area, with around 10 to 20 meters between each node. When the nodes are turned on, they automatically create a network be- tween each other using ESP-NOW technology. This technology allows the nodes to communicate with each other over short distances without the need for Wi-Fi or cellular networks, making it ideal for border areas where such networks may be unavailable or unreliable.

Each node is equipped with a 24GHz mmWave sensor and a GPS module. The 24GHz mmWave sensor can detect moving objects, such as humans and vehicles, at long distances. The GPS module enables accurate location tracking, which is essential for effective border monitoring. The nodes communicate with each other, sending data and location information until it reaches the node closest to the Raspberry Pi, which acts as the base station.



1. ESP32 Micro-controler
2. Solar panel
3. LoRa module
4. GPS module
5. Rechargeable Li-Ion Battery
6. Bottom case
7. Top case
8. Movement sensor

FIGURE 4.4: Complete system outlook

Communication between the last node and the Raspberry Pi is achieved using the MQTT protocol. MQTT is a lightweight and efficient messaging protocol de- signed for use in machine-to-machine (M2M) and Internet of Things (IoT) applications. The use of the MQTT protocol enables the transmission of real-time data from the sensors to the Raspberry Pi, which can then be processed, analyzed, and visualized.

In case a node is far from the rest of the network and cannot send data via ESP- NOW, it can use LoRa technology. The nodes are equipped with LoRa, and they can send data to other nodes equipped with LoRa or directly to the Raspberry Pi. LoRa technology provides redundancy, ensuring that data transmission is not disrupted, and real-time information is always available to monitoring stations and mobile phones.

Each node is equipped with a 5V solar panel for charging the 3.7V lithium battery. This feature ensures that the nodes can work 24/7 non-stop for many years, making them ideal for remote or challenging environments. The nodes also have a tamper protection system that activates a capacitor when the case is opened. This capacitor sends a high current to the chips inside, frying them to protect sensitive data. The system's ability to transmit real-time data to a centralized database and notify monitoring stations and mobile phones of any incidents is an effective approach to border surveillance and monitoring. This capability provides critical information for quick and appropriate action, enabling authorities to respond to incidents in a timely and effective manner.

FIGURE 4.5: Prototype 1

In conclusion, the use of wireless sensor networks equipped with 24GHz mmWave sensors and GPS modules for border surveillance and monitoring is a significant development in national security. The system's use of multiple technologies, including ESP-NOW, MQTT, LoRa, and solar panels, provides reliable and efficient detection of human and vehicle activity, accurate location tracking, redundant data transmission, and long-term operation in remote or challenging environments. The tamper protection system further enhances the system's security, ensuring the protection of sensitive data. Overall, this system provides an effective and comprehensive approach to border surveillance and monitoring, enabling authorities to maintain national security and respond to incidents in a timely and effective manner.

Chapter 5

Software: Base station "Raspberry PI"

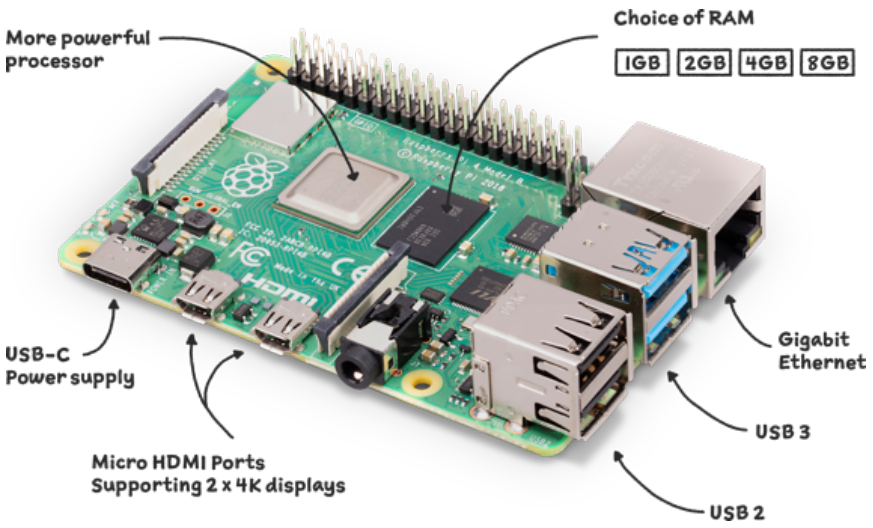


FIGURE 5.1: Raspberry Pi 4 Model B

The Raspberry Pi 4 Model B 8GB is a powerful single-board computer designed for various applications, ranging from hobbyist projects to professional uses. This documentation provides an overview of the key specifications and features of the Raspberry Pi 4 8GB, enabling users to understand its capabilities and make informed decisions when working with this device.

1. **Processor and Memory:** The Raspberry Pi 4 8GB is powered by a Broadcom BCM2711 quad-core Cortex-A72 (ARM v8) 64-bit system-on-a-chip (SoC) with a clock speed of 1.5 GHz. It includes an 8GB LPDDR4-3200 SDRAM, providing improved performance compared to previous models. The increased memory capacity allows for more demanding applications and multitasking.
2. **Connectivity:** The Raspberry Pi 4 8GB offers a range of connectivity options, including:
    - Raspberry Pi OS (previously known as Raspbian), is the official supported operating system.
    - Ubuntu, is a popular Linux distribution.
    - Other Linux distributions, such as Debian, Fedora, and Arch Linux.
    - Windows 10 IoT Core, a version of Windows optimized for IoT devices.
3. **GPIO and Expansion:** The Raspberry Pi 4 8GB features a 40-pin GPIO (General Purpose Input/Output) header, allowing for easy integration with external devices and sensors. This header supports various interfaces, including I2C, SPI, UART, and GPIO, providing flexibility for project development. Additionally, it is compatible with existing Raspberry Pi HATs (Hardware Attached on Top) and accessories.
4. **Power Supply:** The Raspberry Pi 4 8GB requires a 5V USB-C power supply with a minimum current rating of 3A. It is recommended to use the official Raspberry Pi power supply or a reliable third-party alternative to ensure stable and reliable operation.

The Raspberry Pi 4 8GB offers enhanced processing power, increased memory capacity, and improved connectivity options compared to its predecessors. With its

versatile features and compatibility with various operating systems, it serves as an excellent platform for a wide range of projects, including home automation, robotics, IoT applications, media centers, and much more.

5.1 GPS Module

One crucial component integrated into the wireless sensor network (WSN) deployed in mechanical systems is the Global Positioning System (GPS) module. The GPS module serves as a vital source of location information, enabling precise tracking and monitoring capabilities within the network.

The GPS module utilized in the WSN consists of a receiver that communicates with satellites orbiting the Earth to determine the device's geographical coordinates. This receiver leverages the signals transmitted by multiple GPS satellites to calculate the exact position, velocity, and time information. By incorporating the GPS module into the WSN, the network gains the ability to precisely locate individual sensor nodes within the mechanical system.

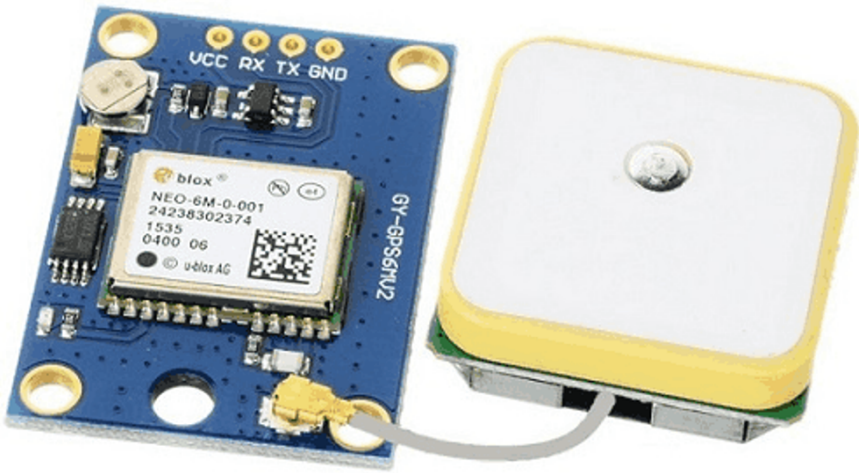


FIGURE 5.2: GPS Module

To incorporate the GPS module into the wireless sensor network, several key steps need to be followed. First, the GPS module should be physically connected to the sensor node, typically through a serial communication interface such as Universal Asynchronous Receiver-Transmitter (UART). This connection allows the sensor node to receive the GPS signals and extract the relevant positioning data.

Once the GPS module is integrated into the sensor node, appropriate software protocols and algorithms must be implemented to process and interpret the received GPS data. These protocols enable the sensor node to decode the satellite signals, extract the location coordinates, and provide accurate positioning information to the WSN.

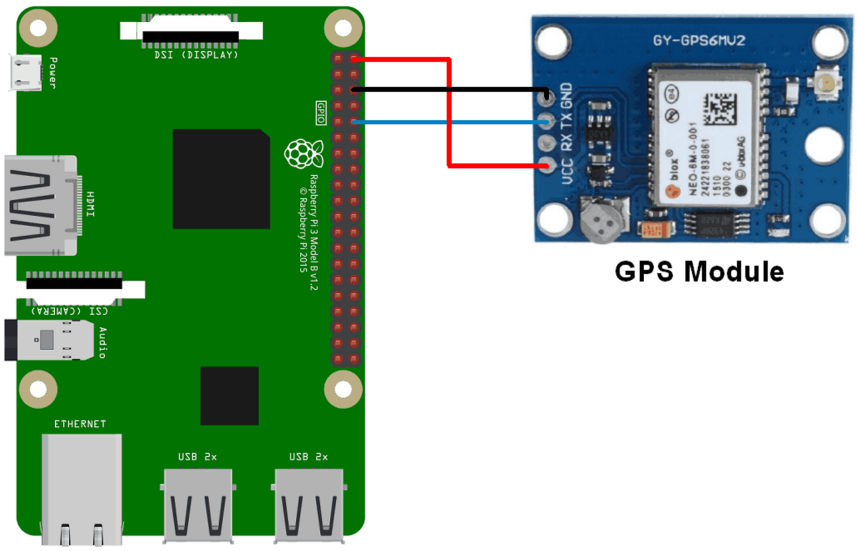


FIGURE 5.3: GPS Module Interfacing with Raspberry Pi

Furthermore, it is essential to consider power management strategies when incorporating the GPS module into the wireless sensor network. GPS modules typically consume more power compared to other components of the sensor node. Therefore, energy-efficient techniques, such as duty cycling or adaptive power control, should be employed to optimize power consumption and prolong the overall network lifetime.

The GPS module in the WSN enables a wide range of applications in mechanical systems. For instance, it facilitates real-time asset tracking, allowing precise monitoring of the movement and location of mechanical equipment or vehicles. Additionally, it enables geofencing capabilities, where specific geographical boundaries can be defined, triggering alerts or actions when the sensor nodes cross those boundaries.

In conclusion, the integration of a GPS module into the wireless sensor network deployed in mechanical systems provides accurate location information and enhances the capabilities of the network. By following the necessary steps for physical integration, software implementation, and power management, the GPS module becomes an invaluable component for precise tracking and monitoring within the WSN

5.1.1 Get GPS Location using Raspberry Pi

We establish a connection between the GPS module and Raspberry Pi to extract GPS information. We can achieve this by interfacing the GPS module with Raspberry Pi using Python and C programming languages (specifically, the WiringPi library for C). To successfully interface the GPS module, follow the connection diagram depicted in the figure provided above, ensuring that the GPS module is properly connected to the Raspberry Pi.

To extract Latitude, Longitude, and time information from the NMEA GPGGA string received from the GPS module using Python, and subsequently print them on the console (terminal), you can follow the steps outlined below. Additionally, with the obtained latitude and longitude values, you can locate the current position on Google Maps.

1. Begin by receiving the NMEA GPGGA string from the GPS module in Python.
2. Parse the received string to extract the relevant information, such as latitude, longitude, and time. This can be accomplished by splitting the string based on the comma-separated values.
3. Retrieve the latitude, longitude, and time values from the parsed string.
4. Print the extracted information on the console (terminal) using the print() function.
5. Utilize the obtained latitude and longitude values to locate the current position on Google Maps by accessing the Google Maps API or by manually inputting the coordinates on the Google Maps website.

By implementing these steps in your Python code, you can successfully extract and display the latitude, longitude, and time information from the NMEA GPGGA string received from the GPS module, as well as locate the current position on Google Maps.

To extract the NMEA GPGGA string and print it on the output window using C programming language with the WiringPi library, you can follow the steps outlined below. Additionally, for more detailed information on how to use the WiringPi library on Raspberry Pi, you can refer to the "How To Use WiringPi Library On Raspberry Pi" resource.

5.1.2 GPS Code for Raspberry Pi using C (WiringPi Library)

```
/*
   GPS Interfacing with Raspberry Pi using C (WiringPi Library)
   http://www.electronicwings.com
*/

#include <stdio.h>
#include <string.h>
#include <errno.h>

#include <wiringPi.h>
#include <wiringSerial.h>

int main ()
{
  int serial_port;
  char dat,buff[100],GGA_code[3];
  unsigned char IsitGGAstring=0;
  unsigned char GGA_index=0;
  unsigned char is_GGA_received_completely = 0;

  if ((serial_port = serialOpen ("/dev/ttyS0", 9600)) < 0)     /* open
   serial port */
  {
    fprintf (stderr, "Unable␣to␣open␣serial␣device:␣%s\n", strerror (
   errno)) ;
```

```
    return 1 ;
  }

  if (wiringPiSetup () == -1)                    /* initializes wiringPi
     setup */
  {
    fprintf (stdout, "Unable␣to␣start␣wiringPi:␣%s\n", strerror (errno)
    ) ;
    return 1 ;
  }

  while(1){

      if(serialDataAvail (serial_port) )      /* check for any data
    available on serial port */
        {
         dat = serialGetchar(serial_port);       /* receive character
    serially */
         if(dat == '$'){
            IsitGGAstring = 0;
            GGA_index = 0;
         }
         else if(IsitGGAstring ==1){
            buff[GGA_index++] = dat;
            if(dat=='\r')
               is_GGA_received_completely = 1;
            }
         else if(GGA_code[0]=='G' && GGA_code[1]=='G' && GGA_code[2]=='
    A'){
            IsitGGAstring = 1;
            GGA_code[0]= 0;
            GGA_code[0]= 0;
            GGA_code[0]= 0;
            }
         else{
            GGA_code[0] = GGA_code[1];
            GGA_code[1] = GGA_code[2];
            GGA_code[2] = dat;
            }
        }
      if(is_GGA_received_completely==1){
         printf("GGA:␣%s",buff);
         is_GGA_received_completely = 0;
      }
    }
    return 0;
}
```

and the **Output** is :

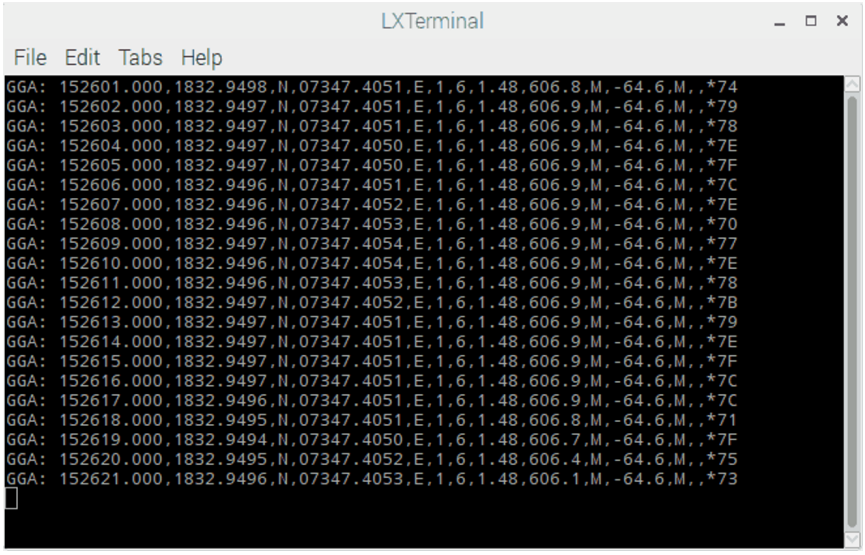


FIGURE 5.4: GPS Output in Terminal

5.2 USB 4G key

USB modem 4G is a type of modem that allows users to connect to the internet wirelessly using a 4G cellular network. It connects to a device, such as a computer or a Raspberry Pi, via a USB port and provides internet connectivity through the cellular network.

When using a USB modem 4G with a Raspberry Pi, there are several things to keep in mind. First, you need to ensure that the modem is compatible with the Raspberry Pi and that you have the necessary drivers installed. Second, you need to ensure that the Raspberry Pi has an adequate power supply to run both the modem and any other connected devices.

FIGURE 5.5: 4G USB Modem

In order to check compatibility, you can refer to the manufacturer's website or documentation. Typically, USB modems with a Huawei chipset are compatible with Raspberry Pi. There are also other modems that are compatible with Raspberry Pi, such as the ZTE MF190 modem. However, it is important to ensure that you have the necessary drivers installed for the modem to work properly with Raspberry Pi.

When it comes to power supply, it is recommended to use a power supply that can provide a minimum of 2.5A for the Raspberry Pi. This will ensure that the Raspberry Pi has enough power to run the modem and any other connected devices.

When using a USB modem 4G with a wireless mesh network, there are additional considerations to keep in mind. Mesh networks are designed to provide reliable and resilient connectivity, but they require careful planning and configuration. It is important to ensure that the mesh network is properly configured to support the USB modem 4G and that the modem is placed in an optimal location to maximize signal strength.

To configure the mesh network, you need to determine the optimal placement of the mesh nodes. The nodes should be placed in a way that provides maximum coverage and minimizes signal interference. You can use tools such as Ekahau or iBwave to plan and optimize your mesh network.

Once the mesh network is properly configured, you need to ensure that the modem is connected to the node that has the best signal strength. This will ensure that you get the best possible internet connectivity through the mesh network.

In summary, USB modem 4G is a useful tool for providing wireless internet connectivity, but it requires careful planning and configuration when used with devices such as Raspberry Pi and in wireless mesh networks. By ensuring compatibility, providing an adequate power supply, and properly configuring the mesh network, you can get the best possible performance from your USB modem 4G.

5.3 Node Red

Node-RED: Node-RED is an open-source flow-based programming tool that provides a visual development environment for wiring together various hardware devices, APIs, and online services. It allows users to create applications and automate workflows by connecting nodes that represent different functions and services. Node-RED simplifies the development process by enabling drag-and-drop programming and providing a wide range of pre-built nodes for different functionalities.

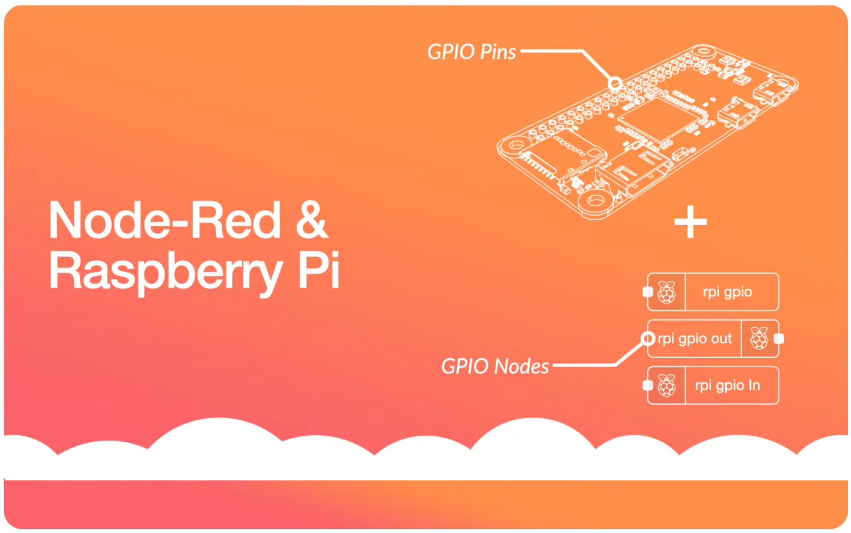


FIGURE 5.6: Raspberry Pi and node red

I'm going to use the Mosquito Mqtt Message Broker Node-red and the Influx Db database. We need a computer to run all of this on, so we're going to use a Raspberry Pi. We're going to install all the different components that we need to set up the data logging system on the Raspberry Pi.

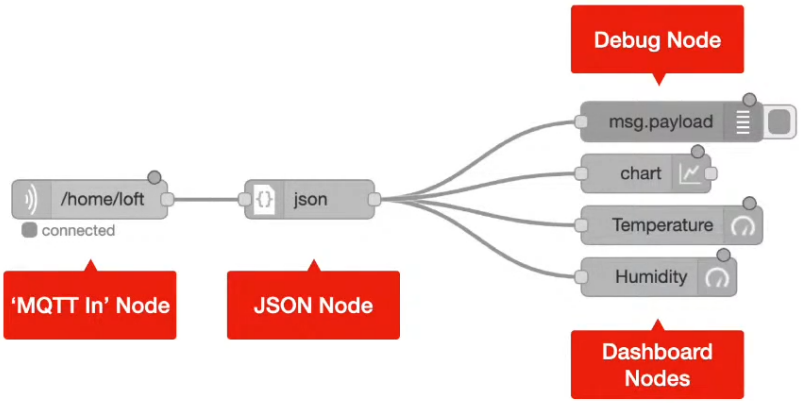


FIGURE 5.7: Node red in our project

Wireless Sensor Networks in Mechanical Networks: The integration of wireless sensor networks in mechanical networks offers several advantages, including:

1. **Real-time Monitoring:** WSNs enable real-time data collection from multiple sensors placed strategically throughout the mechanical network. This facilitates the continuous monitoring of various parameters such as temperature, pressure, vibration, and humidity, providing valuable insights into the system's behavior and performance.
2. **Fault Detection and Diagnostics:** By analyzing the data obtained from the sensors, potential faults or anomalies can be detected in the mechanical network. This allows for timely intervention and preventive maintenance, minimizing downtime and reducing repair costs.
3. **Sensor Node Integration:** Node-RED provides various nodes that enable the integration of wireless sensor nodes with the network. These nodes support different communication protocols such as Zigbee, Bluetooth, or Wi-Fi, allowing seamless integration of sensors into the network infrastructure.
4. **Data Processing and Visualization:** Node-RED offers a range of nodes for data processing, including filtering, aggregation, and statistical analysis. These nodes can be used to preprocess the sensor data before further analysis or visualization. Additionally, Node-RED provides visualization nodes that enable the creation of real-time dashboards, charts, and graphs to monitor the collected data visually.
5. **Event-Based Automation:** Node-RED's event-driven architecture allows for the implementation of automation and control logic based on sensor data. By defining rules and conditions using flow-based programming, it is possible to trigger specific actions or alerts in response to certain events or thresholds being met.

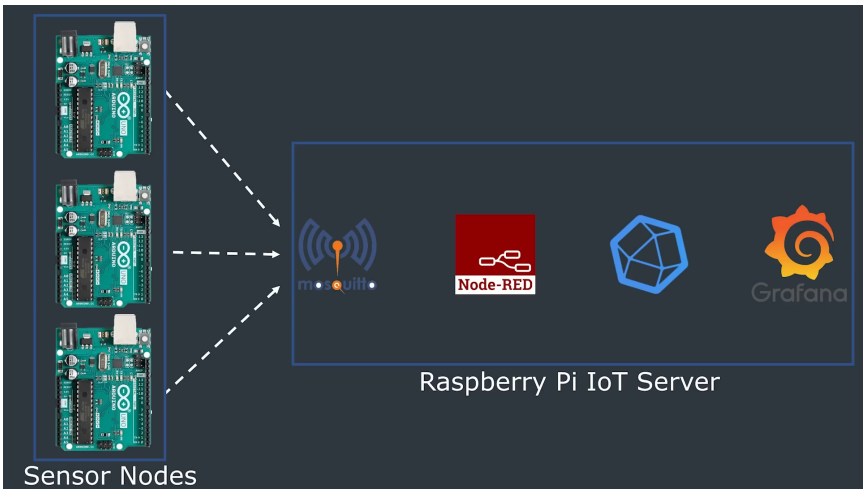


FIGURE 5.8: the plan

The combination of wireless sensor networks and Node-RED provides a powerful platform for monitoring and controlling networks. Through the utilization of Node-RED's visual programming capabilities, seamless integration of sensor nodes, and data processing and visualization functionalities, efficient and effective monitoring of systems can be achieved.

5.4 InfluxDB

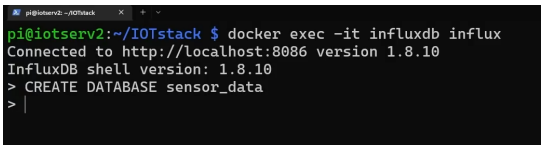


FIGURE 5.9: Database creation

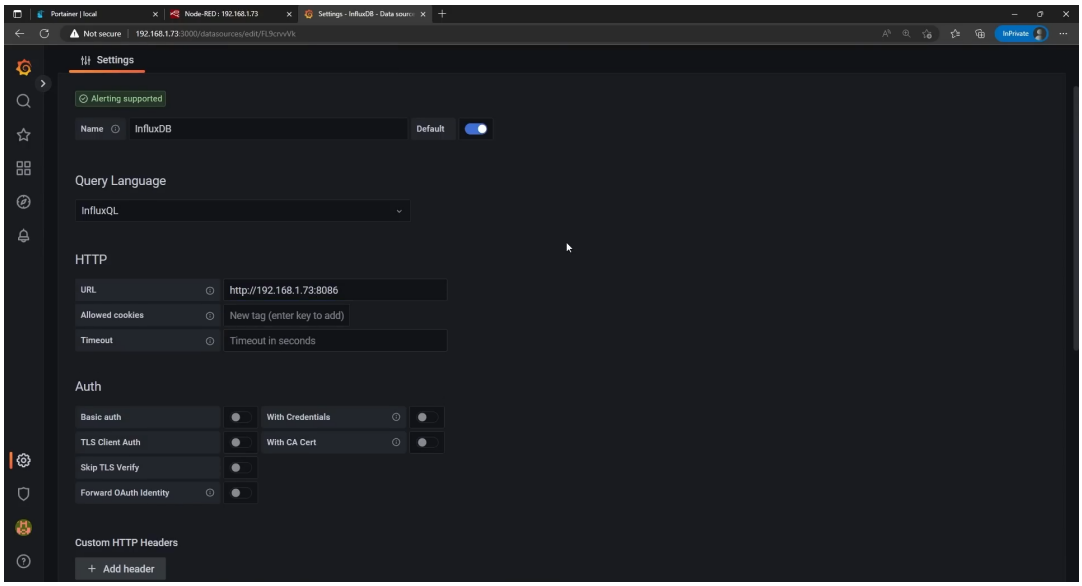


FIGURE 5.10: Database creation 2

InfluxDB, a time-series database, plays a crucial role in wireless sensor networks (WSNs) deployed in mechanical networks. This research focuses on exploring the implementation of InfluxDB as a powerful storage and retrieval system for sensor data collected in WSNs within mechanical networks.

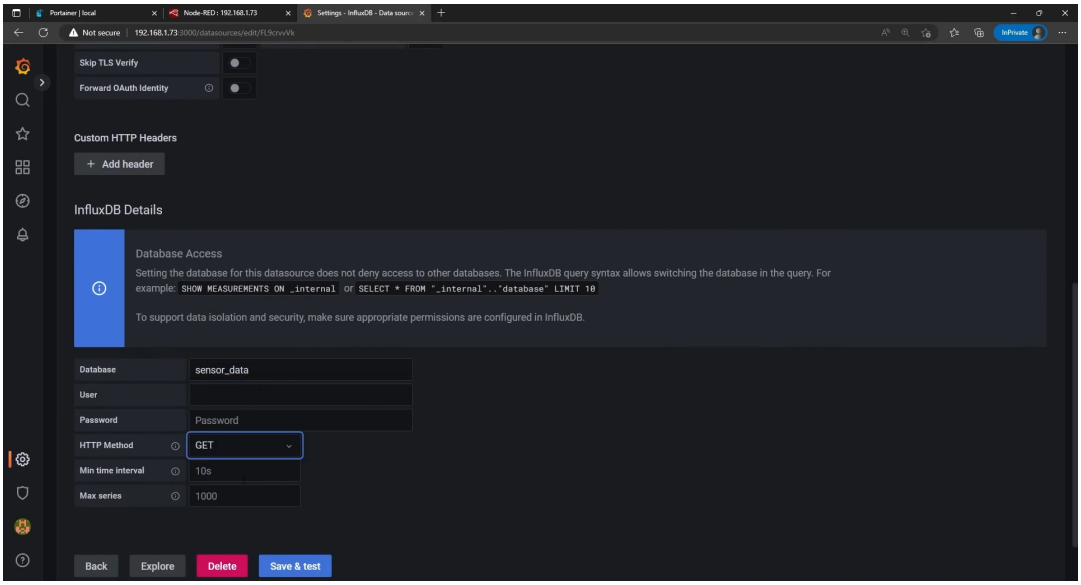


FIGURE 5.11: Database creation 3

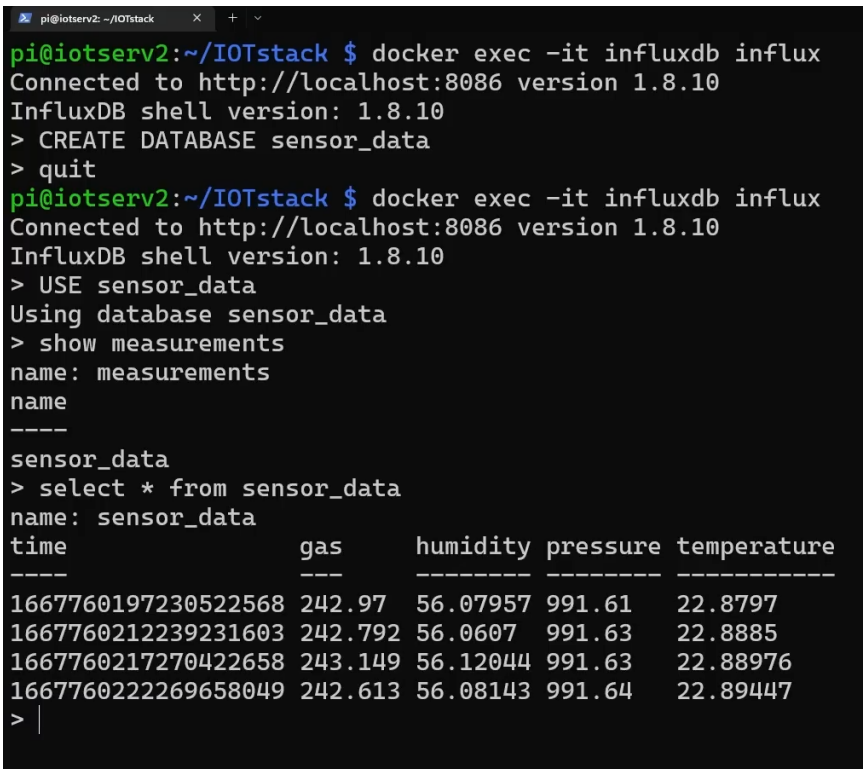


FIGURE 5.12: Database data

InfluxDB's integration into wireless sensor networks provides a robust and efficient solution for storing and retrieving time-series sensor data. By leveraging InfluxDB's capabilities in time-series data management, retention policies, high availability,

scalability, and powerful querying, users can gain valuable insights into the behavior and performance of the mechanical network.

5.5 Docker

The rapid advancement of wireless sensor networks has transformed the field of mechanical engineering, enabling real-time monitoring and data collection from various components within a mechanical network. However, the deployment and management of such networks pose significant challenges, especially when dealing with scalability and reliability. Docker, a containerization platform, offers a promising solution to address these challenges by providing a lightweight and portable environment for deploying and running applications. This research aims to explore the utilization of Docker containers to enhance the efficiency and scalability of wireless sensor networks in mechanical systems.

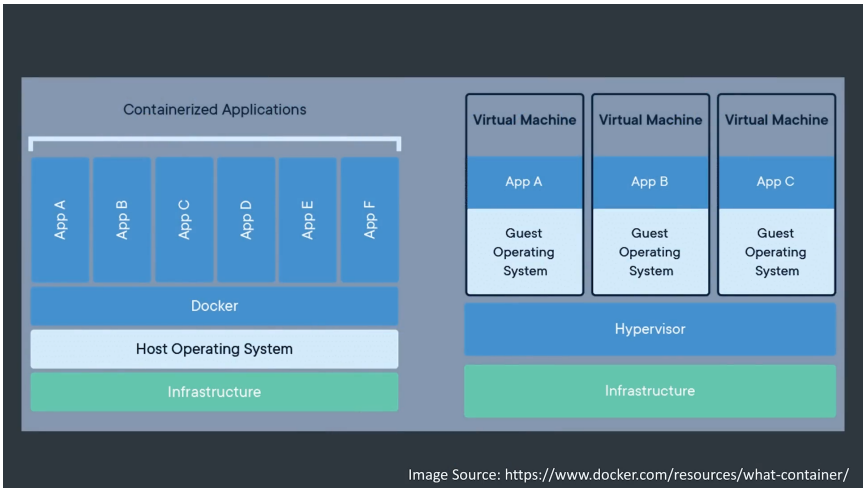


FIGURE 5.13: Docker

Implementing the wireless sensors network using Docker involves the following steps:

1. Designing the network architecture: Determine the structure and components of the wireless sensors network in the mechanical system, including the types of sensors and their communication protocols.
2. Containerizing the software components: Utilize Docker to create container images for the software components required for the wireless sensors network, such as data processing modules and communication interfaces.
3. Developing Docker Configurations: Create Docker files and Docker Compose files to define the specifications and dependencies of the containers. This ensures consistent deployment across different systems and simplifies the management of the network.
4. Deploying the Docker containers: Use Docker to deploy the containerized software components on the target hardware devices within the mechanical network. This allows for easy scalability and distribution of the sensor network.

5. Monitoring and optimization: Implement monitoring mechanisms to track the performance and resource utilization of the Docker containers. Optimize the deployment to ensure efficient utilization of hardware resources.

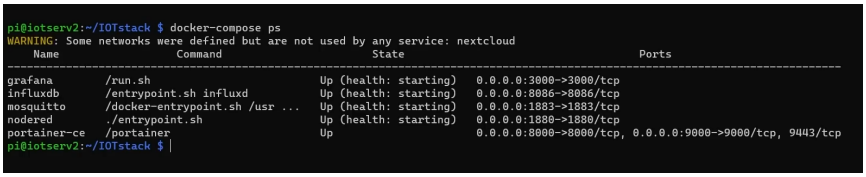


FIGURE 5.14: Docker containers

The implementation of the wireless sensors network using Docker containers provides several advantages. The use of containers allows for easy deployment, management, and scalability of the network. Containers ensure isolation between different components, preventing conflicts and enhancing security. Furthermore, Docker's lightweight nature minimizes resource overhead and allows for efficient utilization of hardware resources. The evaluation of the system demonstrates improved performance, reliability, and flexibility compared to traditional deployment approaches.

The utilization of Docker containers in implementing wireless sensor networks in mechanical systems offers a promising solution to overcome the challenges of deployment and management. Docker's containerization technology provides a lightweight and portable environment, enabling efficient scalability, isolation, and resource utilization. By adopting Docker for wireless sensor networks in mechanical networks, engineers can enhance the efficiency and reliability of monitoring and data collection processes.

5.6 Portrainer

The deployment and management of complex wireless sensor networks in mechanical systems require efficient tools that simplify the administration and monitoring processes. Portainer-CE, a lightweight container management platform, offers a user-friendly interface for managing Docker containers and orchestrating applications. This research aims to explore the utilization of Portainer-CE in enhancing the administration and monitoring aspects of wireless sensor networks in mechanical systems. implementing the wireless sensors network using Portainer-CE involves the following steps:

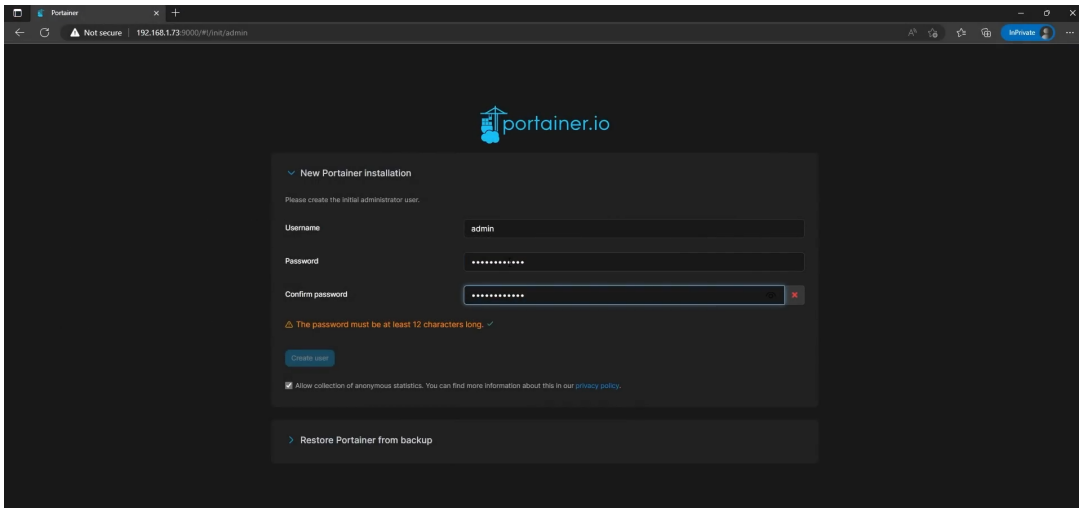


FIGURE 5.15: Portainer

1. Installation and configuration: Set up and configure Portainer-CE on a designated server or device within the mechanical network. Ensure that the server meets the system requirements and establishes appropriate security measures.
2. Docker integration: Connect Portainer-CE to the Docker daemon running on the target hardware devices within the wireless sensors network. This integration allows Portainer-CE to manage and monitor the Docker containers deployed for the network.
3. Container management: Utilize the Portainer-CE interface to manage the lifecycle of Docker containers. This includes container creation, deployment, scaling, and removal. Portainer-CE simplifies container management tasks, allowing engineers to easily handle the deployment and maintenance of containers within the network.
4. Application orchestration: Use Portainer-CE to orchestrate applications within the wireless sensors network. This involves defining and managing multi-container applications using Docker Compose or Kubernetes files. Portainer-CE provides a graphical interface for defining application stacks and managing their deployment and scaling.
5. Monitoring and logging: Leverage the monitoring and logging capabilities of Portainer-CE to track the performance and health of the wireless sensors network. Utilize built-in monitoring tools or integrate external monitoring solutions to gain insights into resource utilization, network traffic, and container metrics.

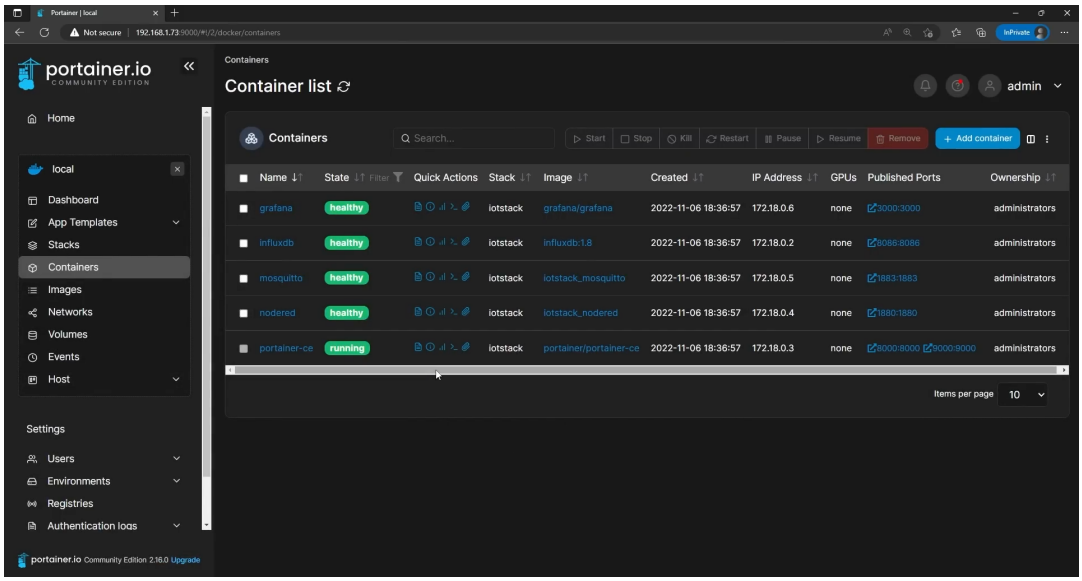


FIGURE 5.16: our containers in portainer

The utilization of Portainer-CE in the wireless sensors network showcases several advantages. The user-friendly interface of Portainer-CE simplifies the administration and management of Docker containers, reducing the learning curve for engineers. The application orchestration capabilities provided by Portainer-CE enhance the scalability and reliability of the network, enabling efficient deployment and scaling of applications. Additionally, the monitoring and logging features offered by Portainer-CE facilitate real-time visibility into the network's performance, aiding in troubleshooting and optimization.

The incorporation of Portainer-CE as a container management platform in wireless sensor networks within mechanical systems enhances the administration, orchestration, and monitoring aspects of the network. By providing a user-friendly interface and powerful management tools, Portainer-CE simplifies the deployment, scaling, and monitoring of Docker containers. This research demonstrates the feasibility and benefits of utilizing Portainer-CE in wireless sensors networks, offering engineers an efficient solution for managing and monitoring complex networks in mechanical engineering applications.

5.7 MQTT protocol

FIGURE 5.17: MQTT protocol

MQTT (Message Queuing Telemetry Transport) is a lightweight messaging protocol commonly used in wireless sensor networks (WSNs) deployed in mechanical networks. This research explores the implementation of MQTT as a reliable and efficient communication protocol for exchanging data between sensor nodes and the central server in WSNs within mechanical networks. MQTT is designed for efficient,

low-power, and low-bandwidth communication in constrained environments. It follows a publish-subscribe messaging pattern, where sensor nodes act as publishers, sending data to the server, which acts as a subscriber. MQTT's lightweight nature and simple protocol make it an ideal choice for resource-constrained wireless sensor networks.

Benefits of MQTT in Wireless Sensor Networks: Integrating MQTT into wireless sensor networks within mechanical networks offers several advantages, including:

1. **Low Overhead**: MQTT has minimal protocol overhead, allowing efficient utilization of network resources. It requires fewer bytes for message transmission, reducing bandwidth consumption and conserving energy in battery-powered sensor nodes.
2. **Asynchronous Communication**: MQTT facilitates asynchronous communication, enabling sensor nodes to publish data whenever new readings are available. This asynchronous nature allows for flexible and efficient data transmission, avoiding the need for constant polling or synchronous communication.
3. **Scalability**: MQTT supports a publish-subscribe model, making it highly scalable for large-scale deployments. Sensor nodes can publish data to multiple subscribers without adding significant overhead, enabling easy expansion of the wireless sensor network as the mechanical network grows.
4. **Quality of Service (QoS)**: MQTT provides different levels of Quality of Service, allowing for reliable message delivery. The QoS levels range from "at most once" (QoS 0) to "at least once" (QoS 1) and "exactly once" (QoS 2). Users can choose the appropriate QoS level based on the reliability requirements of the application.

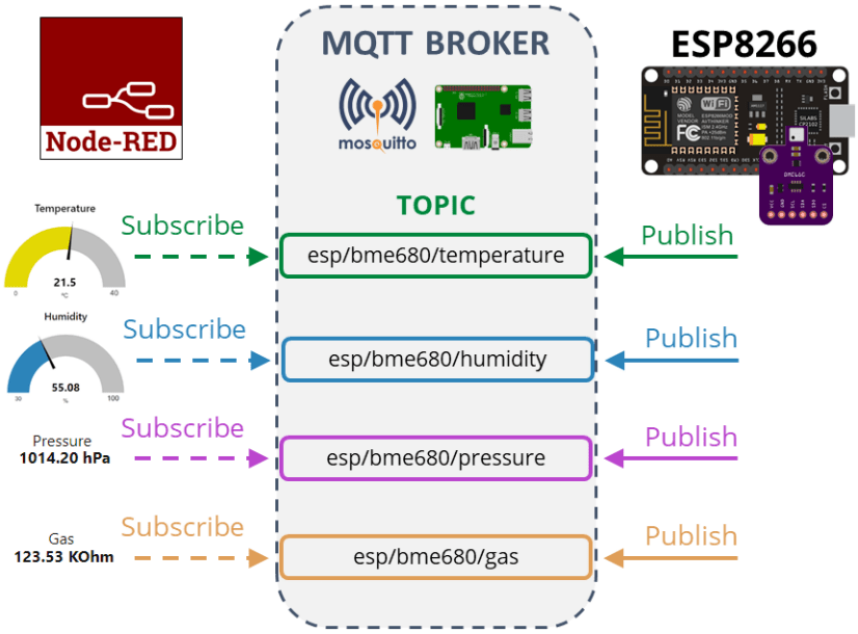


FIGURE 5.18: MQTT and node red

To implement MQTT in wireless sensor networks within mechanical networks, the following steps can be followed:

MQTT Broker Setup: Set up an MQTT broker, such as Mosquitto or Eclipse Mosquitto, which acts as a central messaging hub. The broker handles message routing and ensures that sensor node messages are delivered to the appropriate subscribers.

Sensor Node Configuration: Configure the sensor nodes to act as MQTT publishers. Each sensor node should be assigned a unique identifier or topic, which allows subscribers to differentiate between different sensor data sources.

Server Configuration: Configure the server to act as an MQTT subscriber. The server should subscribe to relevant topics or sensor node identifiers to receive and process the published data from the sensor nodes.

Data Processing and Analysis: Once the server receives the MQTT messages, the data can be processed, analyzed, and stored in a suitable database or forwarded to visualization tools for further analysis and monitoring.

The integration of MQTT into wireless sensor networks within mechanical networks provides an efficient and reliable communication protocol for exchanging data between sensor nodes and the central server. MQTT's lightweight nature, asynchronous communication, scalability, and Quality of Service options make it well-suited for resource-constrained environments. This research contributes to the advancement of wireless sensor networks by highlighting the significance of MQTT in enhancing communication and data exchange in mechanical networks.

5.8 Mosquitto

FIGURE 5.19: Mosquitto

Wireless sensor networks have revolutionized the field of mechanical engineering by enabling real-time monitoring and data collection from various components within a mechanical network. To facilitate efficient communication between these components, a robust and scalable messaging protocol is crucial. Mosquitto, an open-source message broker implementing the MQTT protocol, offers a promising solution for reliable and lightweight messaging in wireless sensor networks. This research aims to explore the utilization of Mosquitto for enhancing the communication capabilities of wireless sensor networks in mechanical systems.

The utilization of Mosquitto in the wireless sensors network showcases several benefits. The MQTT protocol implemented by Mosquitto offers lightweight and efficient messaging, making it ideal for resource-constrained devices in a wireless network. The publish-subscribe model provided by Mosquitto enables scalable and real-time communication between sensors and other network components. Additionally, the secure and reliable nature of Mosquitto ensures the integrity and confidentiality of data transmission.

The utilization of Mosquitto in the wireless sensors network showcases several benefits. The MQTT protocol implemented by Mosquitto offers lightweight and efficient messaging, making it ideal for resource-constrained devices in a wireless network. The publish-subscribe model provided by Mosquitto enables scalable and real-time communication between sensors and other network components. Additionally, the secure and reliable nature of Mosquitto ensures the integrity and confidentiality of data transmission.

5.9 Graphana

Grafana, an open-source platform for data visualization and monitoring, plays a critical role in wireless sensor networks (WSNs) deployed in mechanical networks. This research focuses on exploring the implementation of Grafana as a powerful tool for visualizing and analyzing data collected from wireless sensors in mechanical networks.

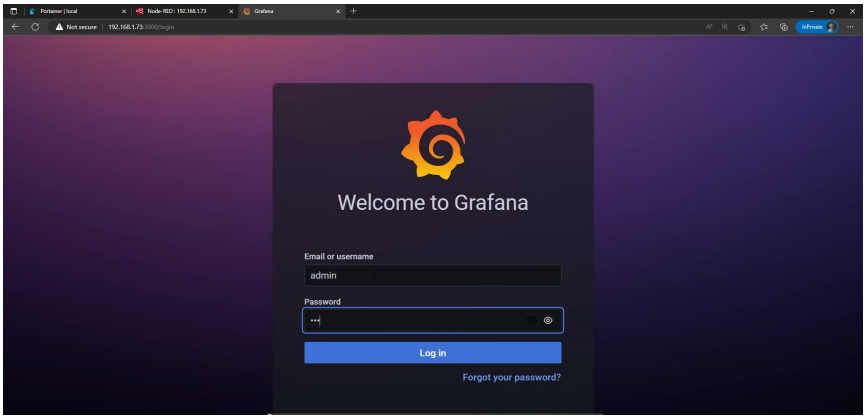


FIGURE 5.20: Graphana

Grafana is a feature-rich platform that provides a user-friendly interface for creating interactive dashboards. It supports various data sources, making it well-suited for visualizing and analyzing data from WSNs. With Grafana, users can create dynamic visual representations of sensor data, enabling better understanding, analysis, and decision-making in mechanical networks.

Benefits of Grafana in Wireless Sensor Networks: The integration of Grafana into wireless sensor networks deployed in mechanical networks offers several advantages, including:

1. **Real-time Visualization:** Grafana enables real-time visualization of sensor data collected from WSNs. Users can create customizable dashboards that display critical parameters such as temperature, pressure, vibration, and humidity, allowing for close monitoring and analysis of the mechanical network's performance.

2. **Data Analysis and Insights**: Grafana supports advanced data analysis features, allowing users to apply statistical functions, perform aggregations, and filter sensor data. These capabilities enable deeper insights into the behavior and performance of the mechanical network, facilitating predictive maintenance, anomaly detection, and optimization.

3. **Alerting and Notifications**: Grafana offers robust alerting and notification features. Users can define thresholds or conditions based on sensor data and receive alerts via email, SMS, or other supported channels when anomalies or critical events occur. This helps ensure prompt response and minimizes potential downtime or failures in the mechanical network.
4. **Data Integration:** Integrate the data collected from wireless sensor nodes into a compatible time-series database or storage system, such as InfluxDB or Prometheus. This ensures that the sensor data can be accessed and queried by Grafana.

Grafana's integration into wireless sensor networks in mechanical networks provides powerful capabilities for data visualization, analysis, and monitoring. With Grafana's flexible dashboard editor, real-time visualization, advanced analysis features, and alerting capabilities, users can gain valuable insights into the performance and behavior of the mechanical network. This research contributes to the advancement of wireless sensor networks by highlighting Grafana's role in enhancing decision-making, predictive maintenance, and optimization in various industrial applications.

Chapter 6

Software: Server and our web application

6.1 The Frontend

The technologies used in the frontend are all of **HTML**, **CSS**, and **vanilla Javascript**. Also we are using b**ootstrap framework** for grid system and web application layout to create a modern and responsive web application.

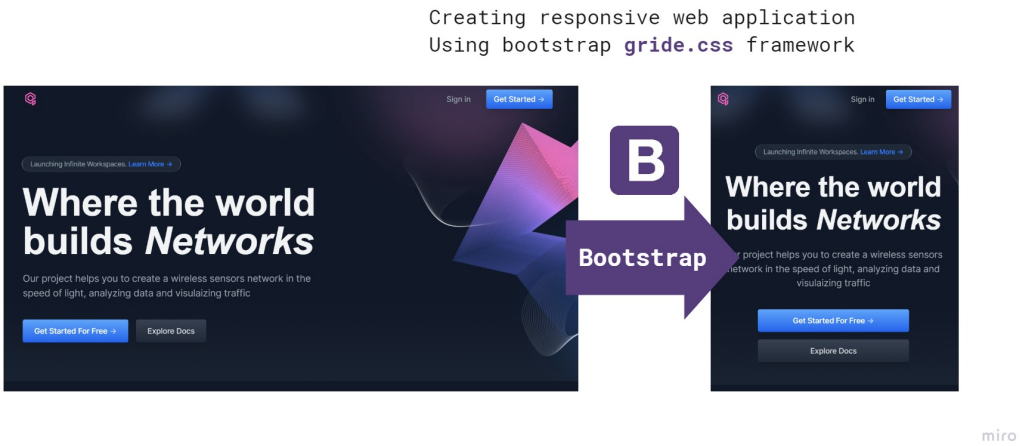


FIGURE 6.1: Responsive web design

6.1.1 HTML (Hyper Text Markup Language)

HTML, which stands for Hypertext Markup Language, is a programming language used to build the structure and content of web pages. It allows you to add tags to text to denote headings, paragraphs, lists, links, and more. HTML is the foundation of web development and is required for the creation of any type of website or web application.HTML is a markup language that structures content on web pages through the use of tags. These tags are used to specify elements on a webpage such as headings, paragraphs, images, links, and lists. Web developers can use HTML to build a logical and ordered framework for the content on a web page, making it easier to read and browse.

HTML is the backbone of any web application or website. It is a necessary component for constructing web pages and is used in concert with other web technologies such as CSS and JavaScript. Web developers can use HTML to construct web sites that are accessible, responsive, and search engine friendly.

HTML has the advantage of being a reasonably straightforward language to learn. Developers may design web pages quickly and easily with just a rudimentary understanding of HTML tags and syntax. Furthermore, HTML is a standardized language, which means that HTML-created web pages may be accessed on any device, independent of platform or operating system.

Overall, HTML is a core web development technology that every web developer should be familiar with. HTML is a key building component that will help you produce organized, accessible, and user-friendly web content, whether you are constructing a basic web page or a complex web application.

6.1.2 CSS (Cascading Style Sheets

CSS (Cascading Style Sheets) is a programming language that is used to style the appearance of web pages. It works with HTML to produce aesthetically beautiful and responsive designs. CSS provides web developers with the ability to control the layout, colors, fonts, and other visual elements of a website or web application.

Web developers can use CSS to generate consistent and appealing designs across many pages and devices. The separation of information and presentation also facilitates web page maintenance and updating, as changes to the CSS file effect the entire site.

CSS has grown over time to include more advanced capabilities like responsive design, which allows web pages to adjust to different screen sizes, and animations, which can improve user interaction and engagement. CSS frameworks like as Bootstrap and Foundation also give pre-built CSS styles and components to help with development speed.

Overall, CSS is a critical technology for developing modern, visually appealing web pages and applications. It gives developers complete control over the style and presentation of content, resulting in a more engaging and user-friendly experience for website visitors.

6.1.3 Vanilla JavaScript

Vanilla JavaScript refers to plain, unadulterated JavaScript that does not make use of any extra libraries or frameworks. It is the most fundamental type of JavaScript and is used to construct dynamic and interactive web pages. Vanilla JavaScript is frequently used to construct web applications alongside HTML and CSS.

Vanilla JavaScript can be used by web developers to provide functionality to web sites such as event listeners, animations, and form validation. Vanilla JavaScript can also be used to interface with server-side technologies like databases and APIs.

The simplicity and ease of use of vanilla JavaScript is one of its key advantages. It is lightweight and fast because it does not rely on any external libraries. It is also more configurable than other JavaScript frameworks, allowing developers to create online applications tailored to their individual requirements.

Overall, vanilla JavaScript is a robust and versatile web development tool. It is a fundamental technology required for creating dynamic and interactive web pages and applications.

6.1.4 Bootstrap version 5

Bootstrap is a well-known front-end web framework for developing responsive and mobile-first websites. It includes a grid structure, pre-built CSS styles, and JavaScript plugins to assist developers in swiftly creating modern and visually appealing online applications.

6.1.5 Javascript Frontend libraries

6.1.5.1 Alpine.js

Alpine is a tough, simple tool for directly building behavior in your markup. Consider it jQuery for the current web. Insert a script tag and get started.

```
<!-- Pricing table 1 -->
<div class="b nl rj ny oe" data-aos="zoom-out">
    <div class="td">
        <div class="oz us tw">Starter</div>
        <div class="o_ nc it tg">
            <span class="ut ui up">$</span>
            <span class="ue uo uf">49</span>
            <span class="ui up">/mo</span>
        </div>
        <div class="up td">For teams building apps for many public &amp;amp; private users.</div>
        <a class="r ud su sl sv fr nj ao fj" href="#0">
            Start Free Trial <span class="uc uv fq ac ah ap tp">-&gt;</span>
        </a>
    </div>
```

FIGURE 6.2: Alpine.js

This figure shows a use of Alpine.js where we are creating a **data-aos** attribute that can be interactive with a css element and javascript to create animation. This animation is used to show a card that zoomin when the user scroll down in the quantum web application landing page. So the user defined **data-aos** attribute is play a very important role in the selection of all the divs as if it's one html element.

6.1.5.2 Chart.js

Chart.js is a well-known JavaScript toolkit that allows you to create flexible and customisable charts and graphs. It provides developers with a simple and user-friendly API for creating many sorts of charts, such as line, bar, pie, doughnut, and radar charts.

Chart.js is built on HTML5 Canvas, allowing it to generate high-performance, interactive charts capable of handling enormous volumes of data. It also provides numerous customization choices, such as changing colors, fonts, and animations, to help you create visually beautiful charts that meet your design specifications.

Chart.js is widely used by developers in a variety of areas, including finance, healthcare, and education, to provide data visualization and reporting. It is an open-source library with a vibrant development community that is constantly updated with new features and bug fixes.

Overall, Chart.js is a robust and adaptable framework for producing beautiful and responsive charts and graphs that can assist you in efficiently visualizing and presenting your data.

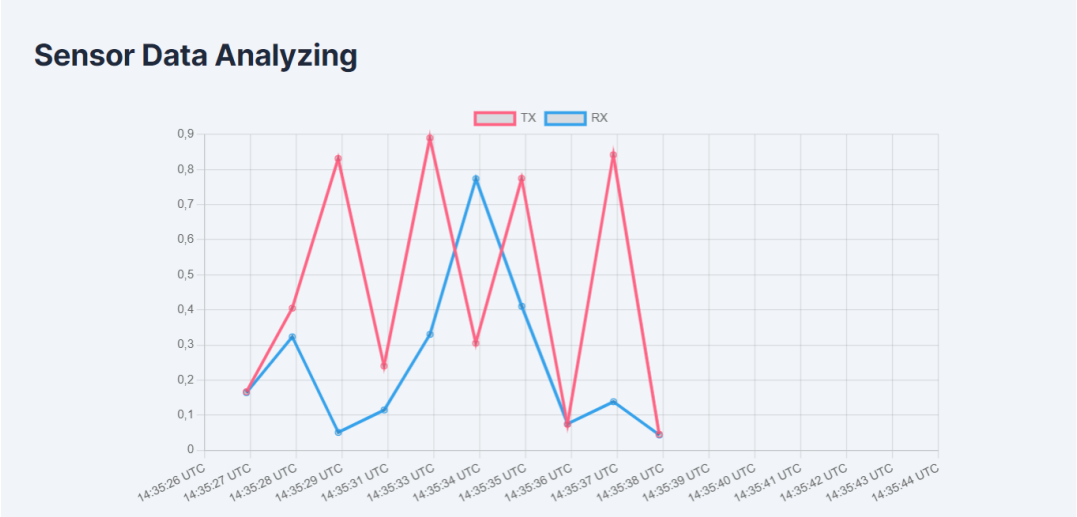


FIGURE 6.3: Sensor analyzing

6.1.5.3 Moment.js

Moment.js is a well-known JavaScript toolkit for manipulating and formatting dates and timings. It has a wide range of functionality, including date and time parsing, validation, and formatting, as well as computing discrepancies between dates and times. Moment.js is lightweight and straightforward to use, with a simple and intuitive API that makes it simple to get started.

The versatility of Moment.js is one of its primary advantages. It supports numerous date and time formats, including ISO 8601, Unix timestamps, and other forms. It also supports numerous locales, which means you may show dates and hours in a variety of languages and formats.

Moment.js is commonly used in online applications ranging from basic date pickers to big scheduling and booking systems. It's also widely used in conjunction with other JavaScript libraries, such React and Angular, to give further date and time features.

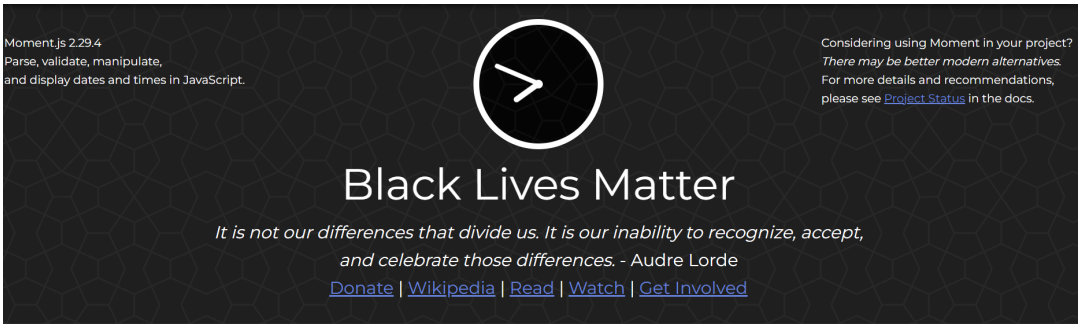


FIGURE 6.4: Moment.js

Overall, Moment.js is a robust and effective JavaScript tool for working with dates and timings. Its adaptability and simplicity make it a popular choice among developers wishing to add date and time features to their web applications.

6.2 Git technology

Git is a software development project management solution that uses a distributed version control mechanism. It enables developers to collaborate on code, track changes, and manage numerous project versions. Git is a valuable tool for software development teams that is extensively utilized in the industry.

Git's distributed nature is one of its primary advantages. Each developer has a complete copy of the project's codebase, allowing them to work offline and merge changes when they are ready. This method makes it simple to cooperate on code and decreases the likelihood of disagreements and errors.

Git also comes with a robust collection of tools for managing code changes. Developers can establish branches to work on specific features or problem fixes, and then merge them back into the main codebase after they're finished. Git also includes code review tools, such as pull requests, which enable other developers to examine and approve changes before they are merged into the main repository.

Another benefit of Git is its adaptability. It is compatible with a wide range of development tools and workflows, and it can be tailored to the specific requirements of a project or team. Git also interfaces with a number of popular development platforms, like GitHub and Bitbucket, which offer extra tools and services for managing software projects.

Overall, Git is a powerful and flexible tool for managing software development projects. Its distributed nature, powerful change management tools, and flexibility make it an essential tool for software development teams.

This is the logs that tracks our web application using **git log** command to print different commit timeline of quantum solution platform.

```
$ git log
commit df45f691e07cd14ca52b23def74a05ee21de47c7 (HEAD -> main)
Author: xbits <xbits@datacopyrightprotection.com>
Date:   Tue Jul 4 06:26:09 2023 -0700

    creating a terminal view connected using ssh

commit 800f4cfcd1d1ecb4083f8f07fda51d63a4beacb8
Author: xbits <xbits@datacopyrightprotection.com>
Date:   Sat Jul 1 12:08:16 2023 -0700

    finishing ssh communication based on random data + deleting useless kit-terminal ejs file

commit 6e5cf425f9c422bf45234cf54eba888a7f84db1e
Author: xbits <xbits@datacopyrightprotection.com>
Date:   Sat Jul 1 06:08:06 2023 -0700

    creating reset route and sensor log route with no official data

commit ac01162b34837dda42df7b3ca067f7da8d093030
Author: xbits <xbits@datacopyrightprotection.com>
Date:   Sat Jul 1 06:07:07 2023 -0700

    creating op route with non official data

commit a43e371454ad71cdae28dbc57595e21d78b6cca6
Author: xbits <xbits@datacopyrightprotection.com>
Date:   Sat Jul 1 01:15:22 2023 -0700
```

FIGURE 6.5: Git

It's offen nice to use sourcetree which is a open source tool that present the project timeline using a cool, modern and well-organized graphic user interface rather than using GIT bash CLI.

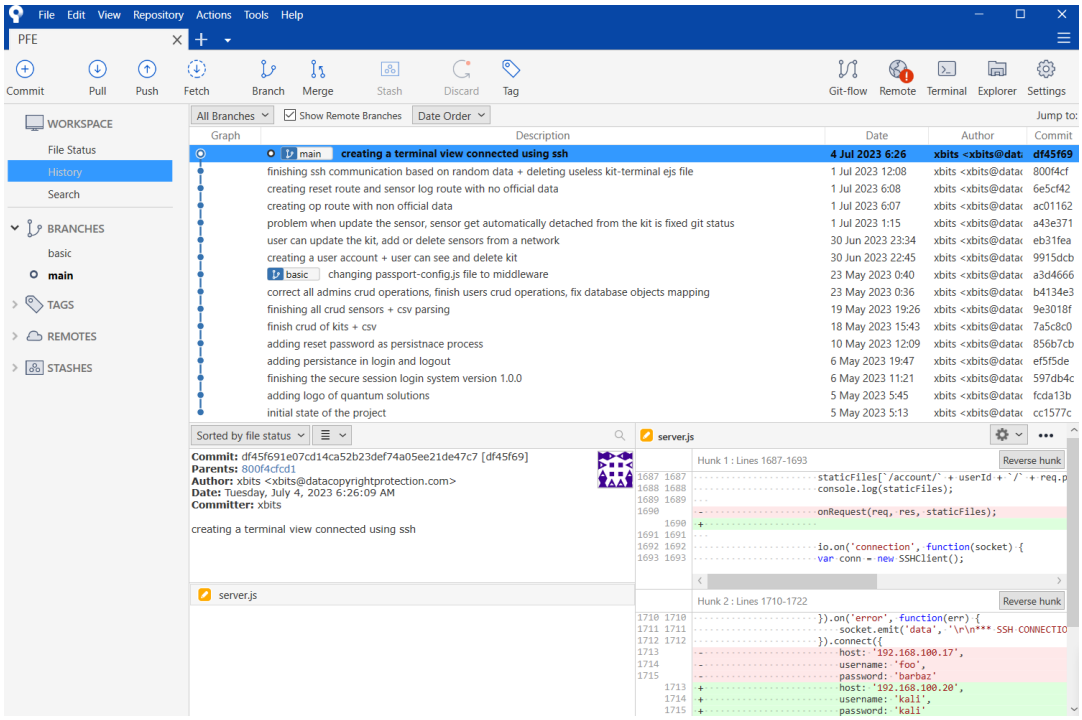


FIGURE 6.6: GUI Git

6.3 API

6.3.1 Definition of an API

An API is a piece of software that allows two apps to communicate with one another. It is a collection of protocols, routines, and tools that govern how software components interact and communicate with one another. APIs are widely used in web development to allow different programs to interact with one another and share data and functionality.

APIs are critical for developing modern software applications because they enable developers to construct modular, reusable code that is readily integrated with other programs. APIs can be used to gain access to data, services, or functionality supplied by other apps or platforms, such as social media platforms, payment gateways, or weather data providers.

APIs are made to be simple to use and to integrate with other apps. They often communicate data in a defined format such as JSON or XML, and they frequently provide a collection of documentation and examples to assist developers in getting started quickly.

APIs are classified into several forms, including web APIs, which are used to access web services or data through HTTP, and device APIs, which are used to access hardware or software functionality on devices such as smartphones or tablets.

APIs also provide a layer of abstraction between distinct software components, which can aid in the simplification and reduction of software program complexity. They can assist developers in creating more scalable, flexible, and maintainable software systems.

Overall, APIs are an important part of modern software development since they allow different programs to communicate and share data and functionality. They provide developers with a strong tool for creating modular, reusable software components that can be readily merged with other applications.

6.3.2 Quantum solution web application api

Our application is based on 3 API's, They are all REST API's build in node.js exchanging JSON files in the communication. The main api it's build to route users using the node.js web server application to a specific end-points depends on the user GET request.

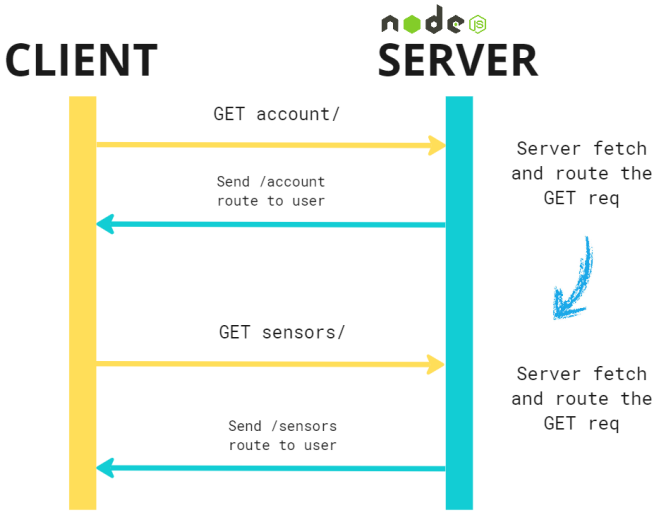


FIGURE 6.7: QS API

The second API is a secure API used to authenticate users devices imidiatly after they powerup them, this was able to acheive using a web tokens based on json-webtoken node package.

You can use an authentication technique such as JSON Web Tokens (JWT) to enable secure and automatic authentication to link each Raspberry Pi to a specific web account on your server. Here is an example of a workflow:

1. Create user accounts on your web server as follows: On your web server, create three user accounts, each with a unique username and password. These accounts will be linked to the appropriate Raspberry Pis.
2. Generate JWT tokens: Implement a way to generate JWT tokens on your web server. Each Raspberry Pi can request a token by supplying its unique identity or credentials when it is turned on. The server validates the credentials and creates a JWT token with the required authentication information.
3. Authentication and data transmission on Raspberry Pi: On each Raspberry Pi, you must develop a script that runs automatically when the device is turned on. This script should carry out the following actions:
    - Obtain the unique identifier or credentials for the Raspberry Pi.

- Send a JWT token request to the server, with the identification or credentials.
- The server will send you the JWT token.
- To transfer data to the server, use the JWT token to authenticate further requests.

4. Data visualization: Once the data is saved on the server, you may use visualization tools to present it in a graphical way. This is possible with web frameworks or libraries like D3.js or Chart.js.

The last API is used to make a communication between the web application running in background with our mobile application. This can be made by creating a react-native mobile application and make it have a GET request to our API that we have published, so each request will be handled in the server side and as result of that we can send data or perform a server side functionalities. Alternative way to that, is to build react-native web-view to make a communication of a mobile device on browser feels like creating a native application the advantage of this solution is that we avoid more traffic comming to the server but there are also drawbacks because it's not a real mobile application and it will not work if the web application doesn't response to different size of devices viewing the app.

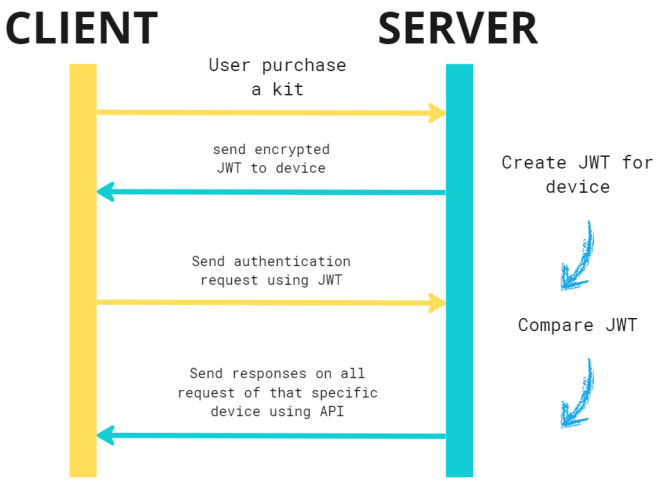


FIGURE 6.8: Client Server

6.4 Backend server

We are using Javascript language to create a web server, to run javascript out of the browser we need to make a new enviroment and api that can help javascript execute backend features, here where node.js runtime enviroment is introduced.

The backend is where the server routes and API creation lives, all security features and authentication procedures and middleware are running on the server side with the bussiness logic which responsible for unicity of quantum web application.

6.4.1 The node.js runtime environment

JavaScript is a programming language that comes within the browser, it's widely used in frontend programming to help achieve website interactivity. JavaScript used an engine to be compiled and then executed by the computer. There are various engines used to execute JavaScript in the browser and one of the most powerful engines in the list is known as the V8 engine. The V8 engine ship within the browser which make it impossible for JavaScript to run outside the browser, we call this a runtime environment, the common runtime environment for JavaScript is to run in the browser using the V8 engine and web API.

Is it possible to run JavaScript outside the browser?

In fact, it's not the browser component that is used to compile and understand JavaScript instructions for our computer but it's the engine inside of it. The extraction of that engine and putting it in another environment can lead to executing and running JavaScript outside the browser and that's the definition of node.js it's a runtime environment for JavaScript used to compile and execute JavaScript instruction directly in the computer which helps us to create backend applications, servers, and utility scripts using JavaScript. In addition, Node.js is a great advantage it came with a new concept in the development industry called full stack development, it's when a developer used to create both frontend and backend programming on his own. It's much easier to perform the same task using the same JavaScript language that runs in different runtimes.

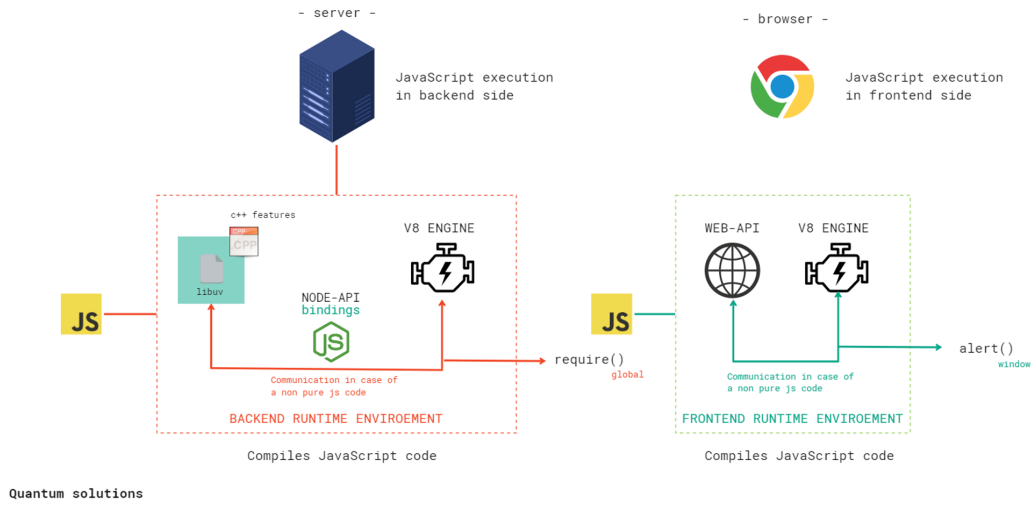


FIGURE 6.9: JavaScript different runtime environments

6.4.1.1 What's a JavaScript Runtime?

A JavaScript runtime is a full environment for reading and executing JavaScript instructions. We cover that node.js is a JavaScript runtime, node.js use a V8 engine written in C++ to execute pure JavaScript functionalities and communicate with the Libuv library if any async code or system feature is introduced. The full work of Openjs was to create the binding between what Google developers create (V8) and what Libuv programmers create where the OS can have a fully functional and secure communication between the V8 and Libuv library.

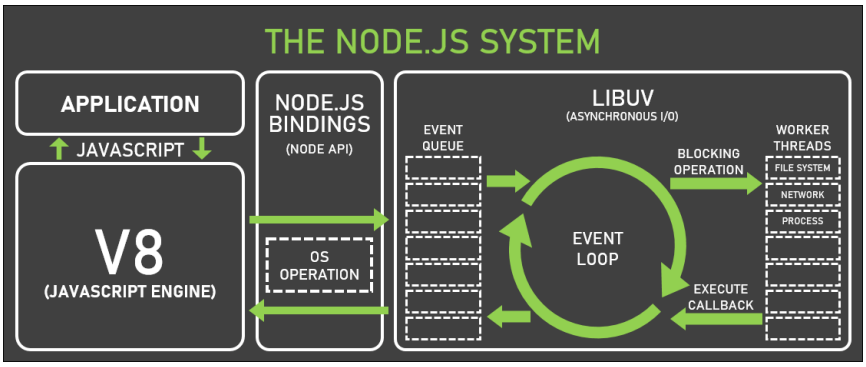


FIGURE 6.10: The node.js system

Those binding are programs that help the V8 engine interface with LIBUV and vice versa, we call that a **node API.**

6.4.1.2 NPM (node package manager)

"Node Package Manager" is abbreviated as NPM. It is a JavaScript package manager that helps developers conveniently obtain and manage dependencies for their Node.js projects.

NPM is used to install, manage, and share packages, which are reusable code blocks that can be used to extend the functionality of your JavaScript applications. These packages might range from simple utility functions to sophisticated frameworks and libraries.

You may quickly install and manage packages with NPM, as well as declare dependencies in your project's package.json file and publish your own packages for others to utilize.

NPM is an essential tool for many developers, particularly those working in the Node.js environment because it makes managing dependencies and sharing code between projects much easier.

6.4.1.3 NVM (node versioning manager)

NVM stands for "Node Version Manager". It is a tool that allows you to easily install and manage multiple versions of Node.js on your computer.

NVM provides an **easy way to switch between different versions of Node.js, so you can test your application on different versions or use different versions for different projects**. With NVM, you can install any version of Node.js and easily switch between them using simple command line commands.

NVM also makes it simple to install and manage global Node.js packages, which may be used in all of your Node.js applications.

NVM is a popular tool among Node.js developers because it allows them to effortlessly handle numerous versions of Node.js while also organizing their development environment.

6.4.1.4 Node.js VS JavaScript

Node.js is a runtime that used a JavaScript engine outside the browser environment. **Node.js is originally based on the V8 Google engine but it can also be possible to run using other JavaScript engines** because in the end all engines are based on Google's open-source V8 engine.

Node.js can run pure JavaScript code with more extra features like the node API in the global object whereas the browser also can run pure JavaScript but it used web API instead to perform extra browser features in the global window object.

```
window.console.log("hello") ===> browser runtime (frontend) client-side
global.console.log("hello") ===> node runtime (backend)     server-side
window. alert()             ===> browser runtime (frontend) client-side
global. process             ===> node runtime (backend)     server-side
```

6.4.2 Frontend and Backend web application

In the web version 2.0, The use of client/server architecture (3-tier architecture) is common, our application is based on the client side which is the interface that runs for our customers, and the backend side which is the business logic behind the quantum solution applications. The MEN stack which stands for (mongo, express, and node) used in our application gives us a full advantage since we use the same programming language in different runtimes, this lead to a fully compatible application which high flexibility. In addition, the V8 engine is always in the support of Google this boost automatically node.js and all the latest version of this runtime knows no gap between the frontend and backend programming interactions.

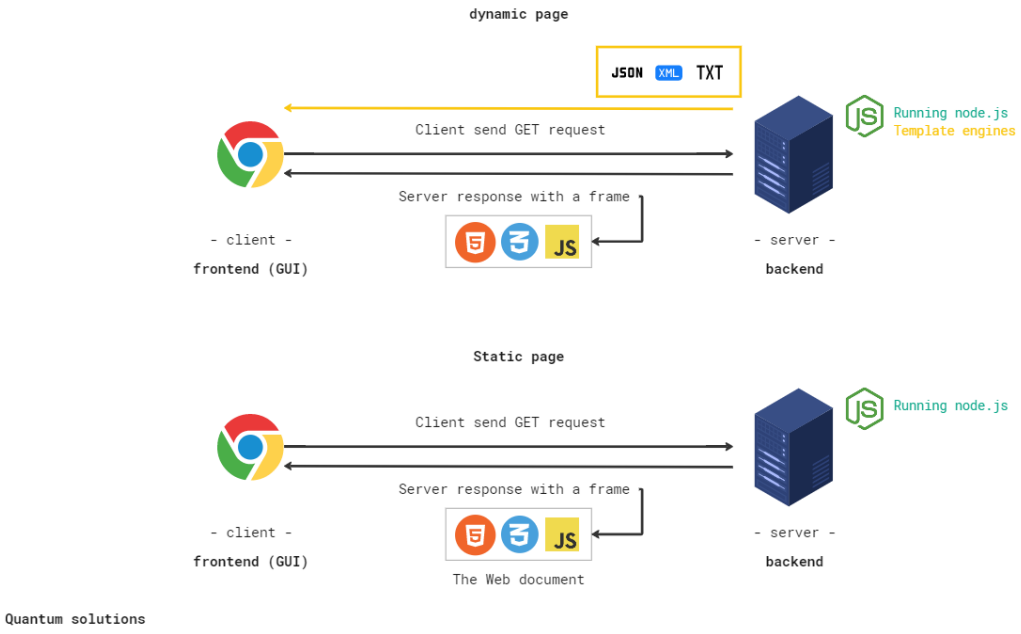


FIGURE 6.11: The Application Architecture

6.4.3 Express.js

Express.js is a popular Node.js web application framework. It provides a simple and extensible API for building Node.js web apps and APIs. Express.js is intended to be lightweight and agnostic, allowing developers to create web applications that are tailored to their individual requirements.

Express.js has a number of features that combine to make it a robust and adaptable online application framework. It contains a strong routing system that allows developers to build routes for managing HTTP requests, as well as middleware for authentication, error management, and other functions. It also has a templating engine, which allows developers to create dynamic HTML sites.

One of the benefits of Express.js is its adaptability. It is a simple framework that does not impose any specific architectural or development style. Developers can build web apps that match their individual needs by combining Express.js with a range of additional Node.js modules and tools.

Express.js is also widely utilized in the business and has a huge and active development community. This community offers a plethora of documentation, tutorials, and examples to assist developers in getting started with the framework and troubleshooting any issues that may arise.

Overall, Express.js is a strong and adaptable online application framework that offers developers a variety of tools for developing modern web apps and APIs. It is a good choice due to its versatility, resilience, and active community.

6.4.4 passport.js

Passport is a Node.js authentication middleware. Passport is extremely flexible and modular, and it can be seamlessly integrated into any Express-based web application. A wide range of authentication methods are available, including login and password, Facebook, Twitter, and others.

FIGURE 6.12: passport code

These are just a few lines that indicate the implementation of authentication using the local strategy with passport.js, the beauty of the framework is that it's lightweight so the developer is the one who can make its own middleware depending on its

need, it just simplifies the communication between middleware developer create to check and authenticate users and user request using Express so the developer skips the routines scripts that may take some times and are not part of a useful creative and unique code. so passports play a significant role when it comes to speeding up the authentication process and authentication persistence using Mongodb communication.

6.4.5 Application main features

6.4.5.1 Application main features

We have successfully developed a highly valuable solution for viewing the network of wireless sensor nodes (WSN) and their corresponding locations based on the transmitting stations using the Google Maps API and JavaScript programming. This feat was made feasible by utilizing the GPS capabilities of Raspberry Pi devices. As each Raspberry Pi collects data coordinates, it sends them to our server in real time. The data is subsequently processed by the server and delivered to the frontend graphical user interface (GUI).

Within this framework, JavaScript dynamically receives data from the Raspberry Pi's GPS module, allowing coordinates to be integrated into the Google Maps API. As a result, Google Maps uses the received coordinates as input and delivers geolocation as output.

We created an effective and efficient system for mapping and visualizing the WSN network using the aforementioned technologies and approaches, allowing for increased insights and informed decision-making.

6.4.5.2 Data analyzing and wsn visualization

We have reached a critical milestone in our development efforts by generating a real-time chart that appropriately depicts the sensor coordinates' frequencies and data logs. This chart is a useful tool for warning users of any detected movement, as well as providing full network representation and monitoring capabilities. We have built a sturdy framework that ensures dependable and precise analysis of the acquired data by utilizing powerful and intricate algorithms.

6.4.5.3 Web shell and ssh communication

Aside from the aforementioned features, our application provides a secure and remote communication route between the user's account and any linked kits. This amazing feature allows users to establish remote connections with each station, allowing them to perform various commands, insert payloads, and run scripts. This brilliant innovation, credited to Hadjazi Mohammed and skilfully implemented by Sahraoui Mohammed, has provided a new degree of flexibility, allowing the application to cater to a wide range of requirements and uses.

By enabling remote interactions with the stations, users get unprecedented control and access, regardless of their physical location. This adaptable flexibility not only improves kit management convenience but also broadens the number of potential use cases. Users can remotely design, monitor, and manage their stations with this remote connection capabilities, giving them increased operational efficiency and flexibility in a variety of settings.

Hadjazi Mohammed and Sahraoui Mohammed's coordinated work have resulted in a cutting-edge feature set that considerably improves the usability and versatility of our program. The remote communication functionality demonstrates their creativity and the application's dedication to provide complete and customised answers to our users' needs.

We would like to express our heartfelt gratitude to Djebbar Yehya for his crucial effort in providing all of the materials required to turn this dream into a practical reality. His passion and drive to obtaining the necessary resources were critical to the effective implementation of our project. Djebbar Yehya's help and generosity were crucial in making our idea a reality, and we truly thank him for his constant work and dedication throughout the process.

6.4.5.4 Drone automated geoposition operations

Introducing our game-changing innovation: drone autonomous geopositioning with an integrated camera system. We created an advanced drone with a high-quality camera that can capture spectacular visual footage while effortlessly executing pre-programmed scripts to move from point X to point Y. Our drone seamlessly maps its trajectory, assuring precise and exact positioning during its aerial voyage by leveraging the power of Google Maps geolocation technology. This ground-breaking solution transforms airborne operations, providing exceptional efficiency, precision, and ease for a wide range of industries and applications. We are pushing the frontiers of what is possible in unmanned aerial systems with our drone automated geoposition operations and integrated camera system, unleashing a new arena of possibilities for collecting magnificent aerial imagery, surveillance, inspection, and much more.

6.4.6 The Application side features

6.4.6.1 Application is asynchronous code based

The application uses wide async operation to give a user a smoother experience while using our web application, in the industry level asynchronous programming becomes the favorite approach can use to exchange data while the core of the application is running. This concept helps the application to jump from single-threaded programming to multi-thread programming performing various tasks at a single point in time.

6.4.6.2 Asynchronous vs synchronous programming

A synchronous code is a code that can be executed instruction by instruction at one point in time, if the t is a task and '**t+1**' is the next coming task in the future '**t+1**' cannot be executed except if t execution and computation are over. It's known as blocking code, the next code cannot be executed unless the current code finishes execution.

An asynchronous code is a code that can be executed instruction by instruction at multiple points in time, if t is a task and '**t+1**' is the next coming task in the future where '**t+1**' doesn't rely on t output in async code we can execute '**t+1**' without waiting for t execution ends and come back later to '**t+i**' instruction that relies upon t when t computation ends. It's known as non-blocking code where the next line gets executed and does not wait for the current line of code to finish its execution. what's

rely on the current line can call later to handle the output using (promises, callbacks, async and wait or fetch).

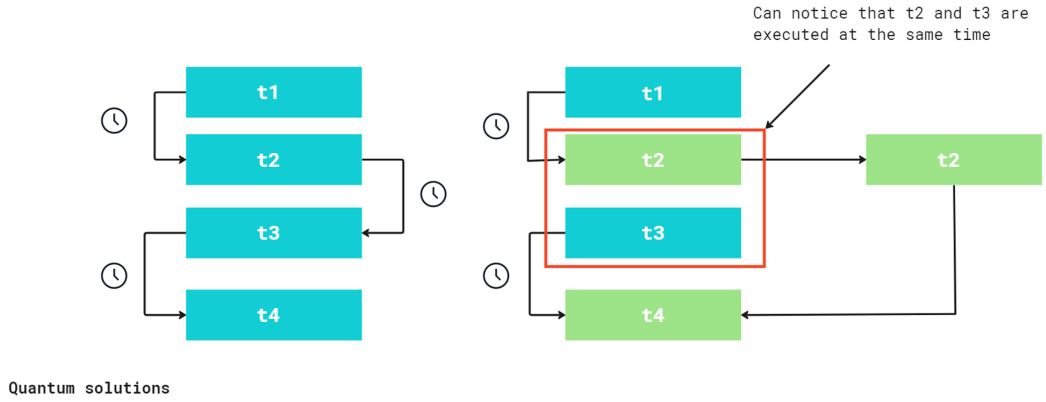


FIGURE 6.13: Asynchronous vs Synchronous

6.4.6.3 Self-build Javascript router

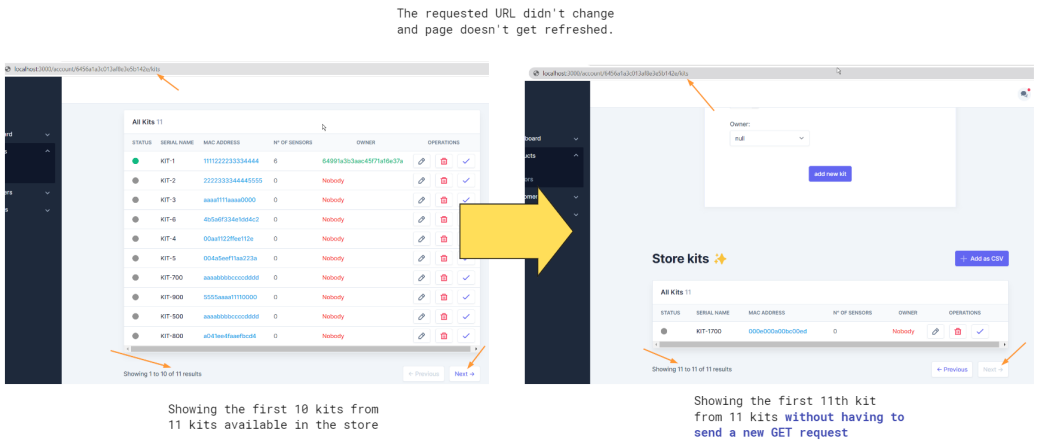


FIGURE 6.14: Javascript router

Instead of using react library that comes with react components and javascript routing functionalities we were able to create our own router to avoid the external react functionalities that our application doesn't require for the moment. We were able to manage pages and list users, kits and sensors comming from database without creating a specific server-side routes, where the user can easly route to the next page without the need refreshing the page hence no GET request is required and this minimize the communication between client-side and server-side hence the user get a smooth experience while using our platform.

Chapter 7

Server Data base

7.1 MongoDB

MongoDB is a frequently used NoSQL database in current web applications. It's a free and open-source document-oriented database that stores information in JSON-like documents with dynamic schemas. MongoDB is a scalable and adaptable solution for managing massive amounts of unstructured data.

MongoDB's strong performance, which is built to enable high write and read throughput, is one of its primary advantages, making it perfect for real-time applications. Furthermore, MongoDB has a document data model, which enables developers to store and retrieve data in a more natural manner than traditional relational databases. This means that developers may work with data in an intuitive framework, which can lead to faster and more efficient development.

Scalability is another feature of MongoDB. MongoDB can manage massive amounts of data and traffic by scaling horizontally across several servers. This makes it an excellent solution for applications that require a big number of users or a huge volume of data.

Another major characteristic that distinguishes MongoDB from other databases is its flexible data model. Data can be added or modified using dynamic schemas without the need for a predefined schema. This makes it simple to iterate on data models and adapt to changing needs.

Finally, MongoDB includes a strong query language that can perform a variety of operations such as filtering, sorting, and aggregation. This makes it simple to extract insights from enormous amounts of data and construct complicated searches to fulfill unique business requirements.

Overall, MongoDB is a robust and adaptable database that can be utilized in a wide range of applications, from small-scale initiatives to large-scale business applications. It is a popular choice among developers and organizations due to its excellent performance, scalability, flexible data model, and rich query language.

7.2 Quantum solutions web application database

In our database, there are three main collections, each of which might include many objects. In this diagram, we can see the relationship between those collections.

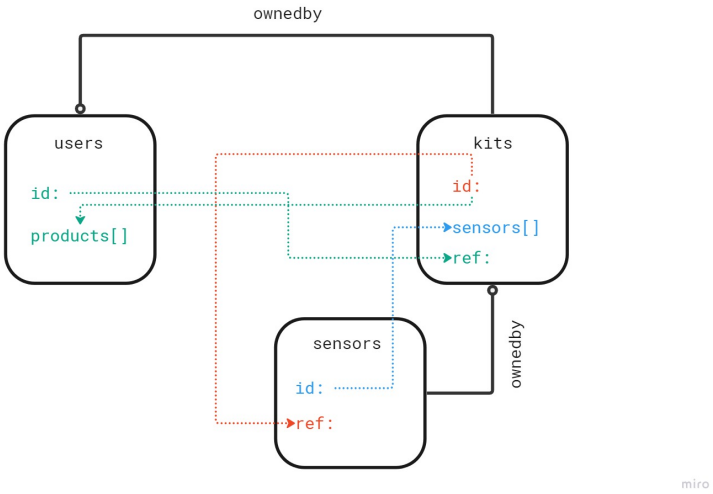


FIGURE 7.1: QS Mongodb schemes

As the graphic demonstrates, the link between sensors and kits is a many to one relationship in which a set of sensors can be owned by a single kit object that represents a station for a wsn network. The same concept applies to users when the relationship between the user's collection and the kits is many to one, since we can easily see that a user can own one or more kits. The sensors attribute, which resides in the kit object, is responsible for holding and containing sensors objects derived from the sensor collection and owned by the kit. Each sensor's ref attribute is mongoose.Object type property that can contain mongoose object ids generated automatically by the mongodb database; in our scenario, the ref of each sensor will have the id of the kit that owns that sensor to maintain database coherence and data intigrity. The similar concept applies to uses and kits, where a user has a products attribute that has the ids of all the kits that the user owns, and all of those kits have a ref attribute that has the value of the id of the user who owns those kits, ensuring data integrity and coherence in the database.

The object kit here is the base object operation; it serves as an intermediary between the collection of users and the collection of sensors; it has two relationships, many to one for users and one to many for sensors. Each kit can have several sensors, but only one person can own them.

7.3 Admin database privilidge

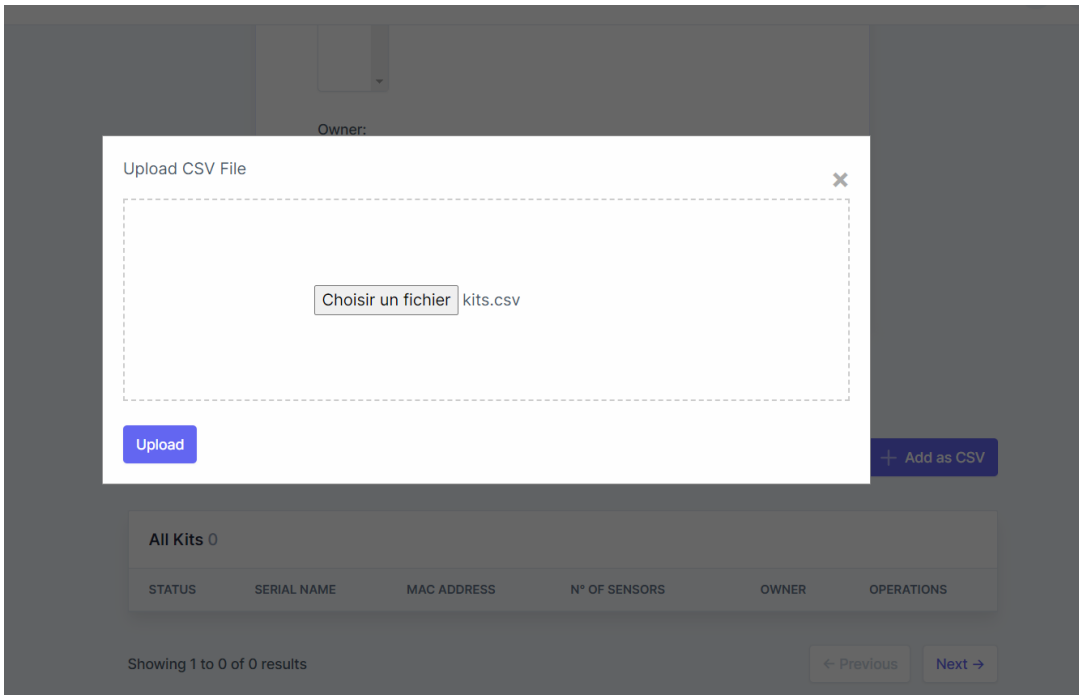


FIGURE 7.2: csv parser

The admin has the most privileges on the Quantium web app and can do all basic operations on users, kits, and sensors objects in the MongoDB. We also provide the ability to use a csv parser to automate the creation of objects with different categories.

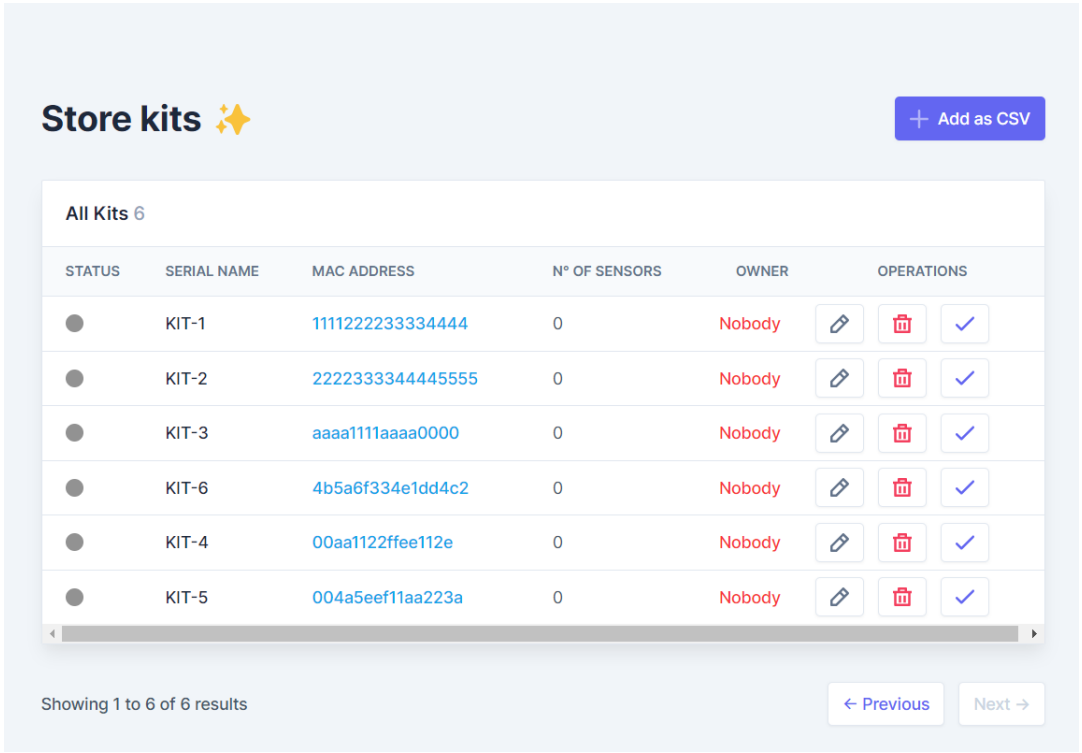


FIGURE 7.3: mongodb objects

After uploading the csv file to the node.js server, it will create a tunnel to receive the

file in a buffer, then treat the file in chunks, parse them to create objects that are compatible with mongodb objects, and save the theme using mongoose commands from the mongoose package, which allows us to apply awesome functionalities between a node server and a mongodb database. This action will result in the instant production of several items, saving administrative time by avoiding the need to create products one by one.

Chapter 8

Mobile App

A mobile app is a computer program or software application that is designed to be used on smartphones and tablets. they are typically downloaded from app stores such as the Apple App Store or Google Play Store, and then installed on the device's operating system. Once installed, the app can be accessed from the user's home screen, and it can be used to perform a variety of tasks, such as playing games, checking email, or accessing social media, in general, the ease of use and the mobility are the main focus here.

8.1 Operating systems for mobile?

We have iOS for the iPhones and then we have Android for all other phones, like Samsung, and Google these are the two major operating systems in the world.But the thing is, you can't take one app, let's say an iOS app and just run it on an Android app, Just like you can't use a Windows application on an apple, this has to do with the underlying architecture that these operating systems are built on.

if you want to build an app, well, you need to use this language called Objective C, which they created, and we can code and build an app that can be put on the App Store. after that google decided to have its own store and named it google app store it also came with its own rules for developing apps in Java as a programing language this are what we call the NATIVE approach of mobile development,and it is also the intended way to do so if we refer to the respective main branches Google and Apple.[int22]

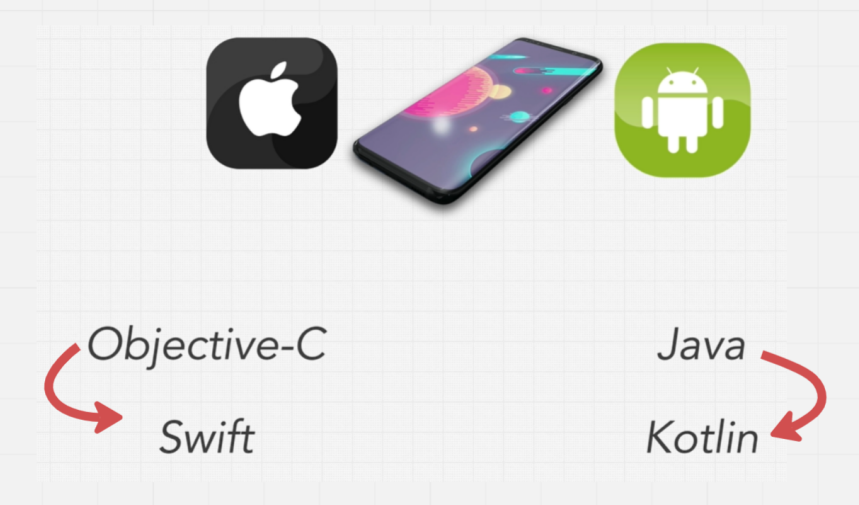


FIGURE 8.1: Mobile development

after the breakthrough of mobile apps and the need for developers in the industry apple decided to change objective c to Swift which is a much easier to understand and work with, google and android too came up with Kotlin which is based on java making it accessible and user friendly for a newcomer to the mobile developing.

8.2 MOBILE DEVELOPMENT NOW:

As we know both the main branches of mobile operating systems has been in search of simplicity and easiness when in it comes to developing their respective app, But the problem is that they present at the same time in the same market competing for

the customer and that implies a choice which is not comfortable zone to be as a developer because when u are developing a mobile app we want to be present on all the side compatible with both systems to get the maximum number of users,or code twice the same app on different systems. and so the people that encouraged mobile development found ways to do exactly that,be present on both systems coding twice.[int22]

8.3 PROGRESSIVE WEB APPs

"Write once, run anywhere."

so the main goal here was to code once while also using a used programing language for the already existing and confirmed web developer, the answer was Cordova and IONIC, well these were the main pillars of the mobile web approach also known as a web view.

so what does this new way bring to mobile development?

1. Cross-Platform Development: One of the primary reasons for using Cordova and Ionic is their ability to develop cross-platform applications. Instead of building separate native apps for each platform (Android, iOS, etc.), developers can write code once using web technologies and deploy it across multiple platforms. This approach saves time, effort, and resources required to develop and maintain multiple native codebases.

2. Web Technology Expertise: Cordova and Ionic leverage web technologies, such as HTML, CSS, and JavaScript, which are widely known and utilized by developers. By utilizing their existing web development skills, developers can quickly transition into mobile app development without having to learn platform-specific languages like Java or Swift. This lowers the entry barrier for developers and promotes rapid application development.

3. Rich UI Components: Ionic Framework provides a wide range of pre-designed and customizable UI components, including buttons, forms, menus, navigation patterns, and more. These components are designed to mimic native app elements, resulting in visually appealing and intuitive user interfaces. By utilizing these components, developers can create polished and engaging mobile apps with less effort compared to building UI elements from scratch.

Overall, the need for using Cordova and Ionic arises from their ability to facilitate cross-platform development, leverage existing web technology expertise, provide rich UI components, access native device features, and offer simplified development workflows. These frameworks empower developers to create mobile applications efficiently and bridge the gap between web and native app development.[rea]

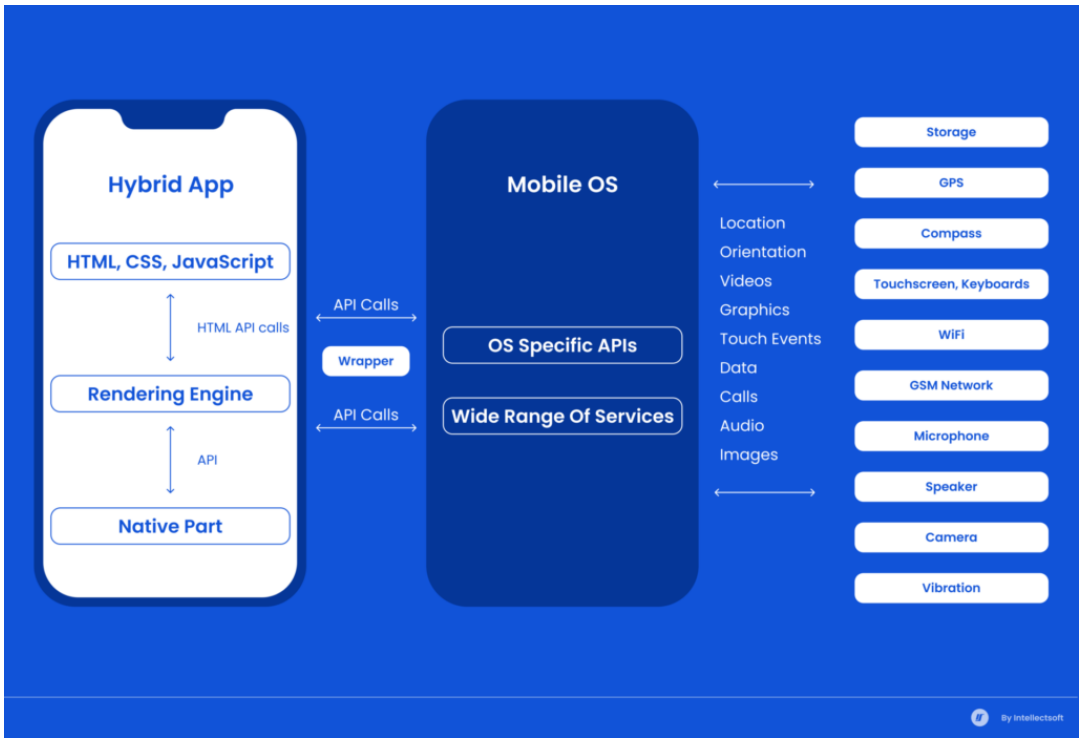


FIGURE 8.2: Hybrid Mobile App Architecture

If the progressive web apps have solved the problem of cross platforming it has also revealed many flaws that the native approach has covered over the past:

1. Performance limitations: Cordova and Ionic rely on a WebView to render the user interface of the application. This can introduce performance overhead compared to native applications, as WebView needs to interpret HTML, CSS, and JavaScript code. Performance can be a concern for applications that require heavy graphics or intense processing, as the WebView may not be as efficient as native code.
2. Limited native functionality: While Cordova provides access to native device capabilities through plugins, not all device features may have plugins available. In some cases, developers may need to create custom plugins to access specific functionality, which can be time-consuming and require additional expertise.
3. App store restrictions: While Cordova and Ionic applications can be deployed to app stores, some app stores may have specific restrictions or guidelines that can affect the submission and acceptance process. It's important to familiarize yourself with the guidelines of each app store and ensure compliance.

8.4 NATIVE-LIKE APPROACH

We know that native mobile apps are really performing really well, but you almost need two separate teams to create an app, Web was great because you only had to write it once and it worked on both phones. However, it just didn't have that good performance, Then we have native-like tools.

this new way of developing mobile app has proven itself in the market performance wise and getting past limitation wise, Microsoft started it with xamarin it was a way to write iOS mobile apps and Android apps using the dominant framework created using C sharp language. [doc23]

the main advantage it has is that it compiles code to native. So we write the code in C Sharp and then it converts that code composite to native code.

now it does more than that but essentially you write the code of your app in other languages depending on the framework you use then it compiles itself to native so that it works on both systems, and the main and most successful framework that really took off the market was React NATIVE, withe react native we can have native views of the app the intended way by the two systems and that already has improved the performance comparing it to the hybrid approach.

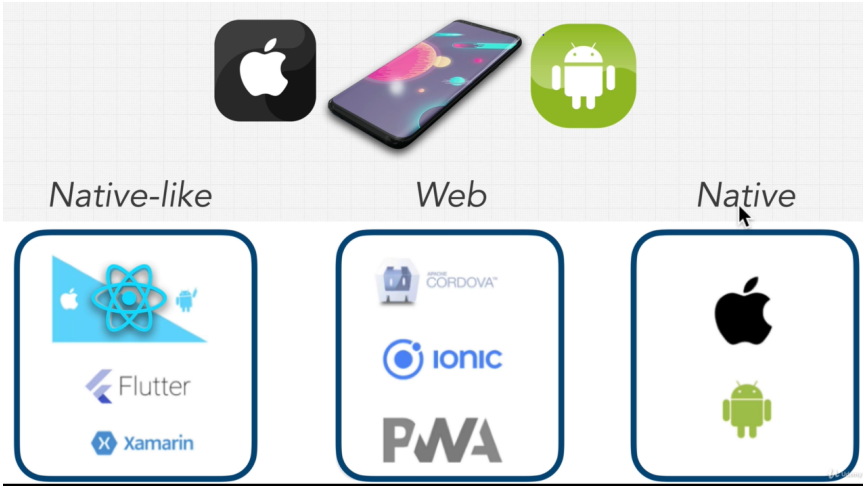


FIGURE 8.3: Mobile Development Options

8.5 React Native

"Learn once, write anywhere."

React Native is an open-source mobile application framework created by Facebook for building cross-platform mobile applications using JavaScript and React. It allows developers to build mobile apps for iOS, Android, and other platforms using a single codebase, instead of having to develop separate apps for each platform. it uses a similar architecture to React, where developers build UI components using JavaScript and a virtual DOM, allowing for a more efficient and responsive user interface. It also includes built-in components and APIs that allow developers to access device features like the camera, accelerometer, and more.[doc23]

React Native has gained popularity among developers due to its ability to save time and resources by allowing them to build cross-platform apps with a single codebase. It is widely used in mobile app development and has been adopted by companies like Facebook, Instagram, and Airbnb.

8.6 Why React Native:

The selection of React Native as the primary technology for developing the application stems from its inherent advantages and compatibility with the objectives of this study. React Native, an open-source framework has gained prominence in recent years for its ability to facilitate the creation of cross-platform mobile applications using a single codebase. This characteristic provides numerous advantages, particularly for academic research endeavors seeking to maximize efficiency and reach a wider user base. By utilizing React Native, the app developed for this study can be seamlessly deployed on both iOS and Android platforms, thereby ensuring broad accessibility and compatibility with a diverse range of devices. Additionally, React Native's component-based architecture fosters modularity, code reusability, and rapid development cycles, which are highly desirable features for an academic project with a predetermined timeline. The versatility of React Native further allows for efficient debugging and maintenance, crucial aspects for long-term sustainability and future enhancements. Overall, the selection of React Native aligns with the objectives of this study, offering a robust and efficient framework for developing a cross-platform mobile application while streamlining the development process and ensuring broad access to the target audience.[doc23]

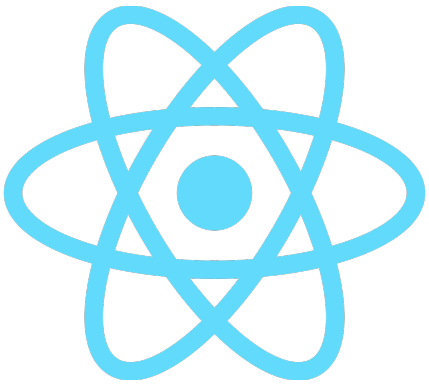


FIGURE 8.4: React Native

The objective of the next chapters is to shed light on the process, decisions, and technologies we used to build the mobile app component, focusing on React Native and Expo. In this memoir, we will explore our development environment, the rationale behind our technology choices

8.6.1 Development Environment:

Our development environment is a crucial foundation for building the mobile app. We decided to use Android Emulator for testing purposes, ensuring compatibility with a wide range of Android devices. The advantage of using an emulator is that it provides a simulated environment that allows us to test the app on various screen sizes and configurations without requiring actual devices. We used VS Code as our integrated development environment (IDE) due to its extensive plugin ecosystem, making development more efficient and enjoyable.[cod]

8.6.2 Choosing React Native:

After an extensive analysis of different frameworks, we unanimously agreed on using React Native for building our mobile app.

The decision to use React Native was driven by its robustness, popularity, and large community support. Additionally, since we had already utilized React for the website, choosing React Native allowed us to leverage our existing knowledge and codebase, streamlining development and reducing potential roadblocks.[doc23]

8.6.3 The Significance of Expo:

To further enhance our development experience with React Native, we embraced Expo, a set of tools and services that simplifies and accelerates the React Native development process. Expo provides a wide range of pre-built components, APIs, and a development server, making it easier to implement complex features without dealing with the intricacies of native modules.[doc]

One of the most significant advantages of Expo is its "Over The Air" (OTA) updates, allowing us to make changes to the app without requiring users to download a new version from the app store. This feature is invaluable for quick bug fixes and feature improvements. Additionally, Expo enables us to easily share our work with stakeholders for testing and feedback, facilitating a smoother collaboration process.

FIGURE 8.5: expo

8.6.4 Collaborative Development Process:

Throughout the development process, we adopted an agile approach, conducting regular sprints to review progress, discuss challenges, and set new goals. We made use of version control systems like Git to ensure seamless collaboration and code synchronization between team members. Code reviews were conducted regularly to maintain code quality and identify potential issues early on.[ldp]

8.6.5 Web View Component?

The decision to incorporate a web view component was driven by several factors. First and foremost, it allows us to leverage the existing website's functionality and content without the need for extensive re-coding or duplicating efforts. This approach enables us to save development time and resources while ensuring a consistent user experience across platforms.

Additionally, utilizing a web view component allows us to tap into the benefits of responsive web design. With responsive web pages, we can adapt the content and layout to different screen sizes and orientations, guaranteeing optimal viewing experiences for users on various devices. This adaptability enhances user satisfaction, as they can enjoy the same rich content and functionality they are familiar with from the website.[blo23]

The Library: React Native WebView To implement the web view component in our mobile app, we turned to the React Native WebView library. This library provides a comprehensive set of tools and APIs that enable seamless integration of web content into our React Native app.

React Native WebView offers a range of features that enhance the user experience and provide flexibility for developers. It supports loading web pages from local or remote sources, allowing us to easily render web content within our app. Additionally, it provides methods for controlling navigation, handling events, and injecting JavaScript code into the web view, expanding our possibilities for customization and interactivity.[rea]

By using React Native WebView, we were able to achieve a high level of integration between our app and the website. This integration fosters a cohesive user experience, enabling users to access and interact with the web content seamlessly as if it were a native part of the app. [blo23]

FIGURE 8.6: webview

In conclusion, the decision to incorporate a web view component in our mobile app was driven by the desire to seamlessly integrate the existing website and provide users with a consistent and engaging experience. By leveraging the power of React Native WebView, we were able to effortlessly render web content within our app and offer users the familiar functionality and responsiveness they expect. The use of the web view component not only enhances user experience but also saves development time and resources, making it a valuable addition to our mobile app.

Chapter 9

Drone Addition

9.1 Why use a drone

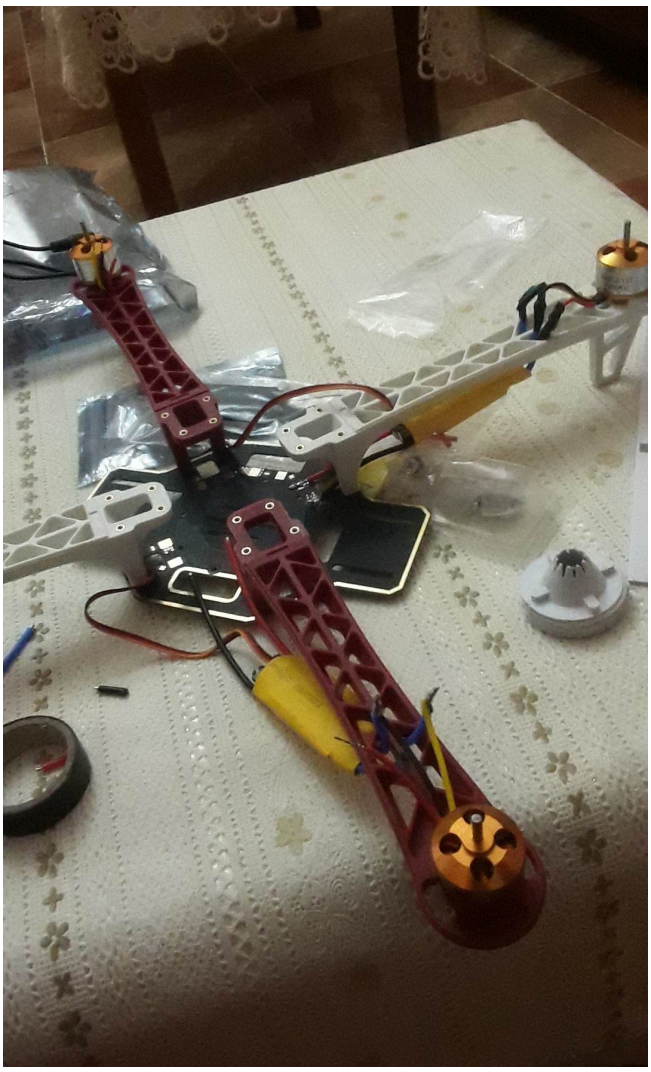


FIGURE 9.1: Drone build

Unmanned aerial vehicles (UAVs), commonly known as drones, have become increasingly popular for various applications, including autonomous flight experiments and aerial surveillance. This tutorial aims to provide a detailed walkthrough of building an ESP32 drone that can be controlled using a web browser interface. The motivation behind this project stems from the limited functionality of existing flight controllers and the desire to leverage web development skills to establish a connection between the drone and the internet, enabling control commands through higher programming languages such as JavaScript.

It is essential to note that drones are potentially dangerous machines capable of causing significant damage to property and posing risks to personal safety. Any attempts to construct and operate a drone should strictly adhere to relevant laws and regulations governing unmanned aircraft systems in the respective jurisdiction.

9.2 Selection and Construction of a Suitable Drone:

Before delving into the specifics of implementing the web-controlled functionality, it is crucial to select and construct a suitable drone. While this tutorial does not focus on the construction process itself, certain insights and considerations can be shared. The initial drone used in this project was a QAV 250 kit. However, it proved unsuitable due to its high speed, limited payload capacity, and asymmetric stability issues. Consequently, an alternative setup was adopted for the ESP32 drone implementation, consisting of a F450 frame, GARTT ML 2212 920KV 230W motors, 10x4.5" propellers, and an Asgard32 F7 flight controller with ESC. Various 3S and 4S batteries, such as CNHL 4S 1500mAh, were used to power the drone.

FIGURE 9.2: moter

FIGURE 9.3: motor 2

9.3 Selection and Construction of a Suitable ESP32 Module:

The ESP32 module acts as the brain of the drone, facilitating the web browser control functionality. The Wemos D1 ESP32 module, lacking an external antenna connector, was chosen for this implementation. While it has a range of approximately 50 meters in an open space without an external antenna, it is important to note that this range was measured using a Samsung Galaxy S6 smartphone. To ensure a stable power supply for the ESP32, an LF33CV regulator was utilized. The Wemos D1 R32, with its onboard power supply, was also considered but did not function optimally with the Asgard32 F7 board due to voltage compatibility issues.

9.3.1 Asgard32 F7 BetaFlight Setup:

To optimize the flight controller's performance, it is crucial to update the Asgard32 F7 firmware to the newest version (AG3XF7). Linux users must allow write access to USB, following the instructions provided in the BetaFlight GitHub repository. After updating the firmware, various configuration options need to be checked, including serial receiver configuration, ESC motor features (e.g., DSHOT1200 and MOTOR STOP), board and sensor alignment, receiver settings, PID tuning, and mode configuration.

9.3.2 ESP32 Software Upload:

To enable the web-based control of the ESP32 drone, the corresponding software needs to be uploaded to the module. The project's source code repository can be cloned using the following command:

Two additional files are required to compile the ESP32 project successfully. The first one is "secret.h," where the drone's Wi-Fi SSID and password can be specified. By default, the SSID is set as "quadcopter" with the same password. The second file is the JS controller, which can be copied from the rc-controller project or linked symbolically.

This comprehensive guide has introduced the process of building an ESP32 drone controlled through a web browser interface. While emphasizing the importance of safety and regulatory compliance, the tutorial covered various aspects, including the selection and construction of a suitable drone, the integration of an ESP32 module, the BetaFlight setup, and the software upload process. By leveraging web development skills and the capabilities of the ESP32 module, drone enthusiasts can explore the exciting possibilities of web-controlled UAVs. It is crucial to remember that responsible operation and adherence to legal requirements are paramount to ensuring safe and secure drone usage.

9.4 ESP32-CAM

In addition to the web-controlled functionality, the surveillance capabilities of the ESP32 drone can be further enhanced by incorporating an ESP32-CAM module to capture high-resolution video footage of the targeted areas. The ESP32-CAM module integrates an ESP32 chip with a camera module, allowing real-time video streaming and recording. By strategically positioning the drone and activating the camera module, it becomes possible to obtain valuable visual data for surveillance purposes.

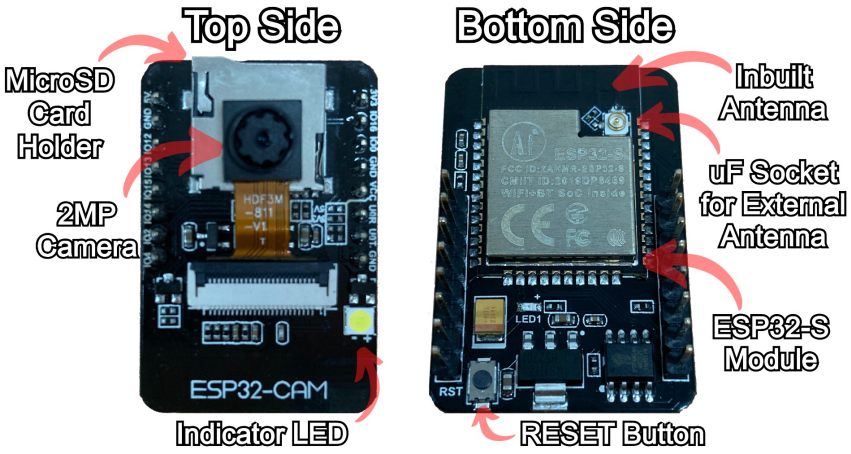


FIGURE 9.4: ESP32-CAM

The ESP32-CAM module offers various features that facilitate effective video capturing. It supports different resolutions and frame rates, enabling the selection of optimal settings based on the surveillance requirements. Additionally, the module allows for the configuration of image settings such as brightness, contrast, and saturation, ensuring clear and detailed video footage.

The integration of the ESP32-CAM module with the drone's web-controlled interface provides operators with the ability to remotely control the camera's functions. Through the web browser interface, users can initiate video recording, capture images, adjust camera settings, and stream live video feeds directly to the base station for real-time monitoring and analysis. This capability enhances situational awareness and enables authorities to respond promptly to potential threats or suspicious activities.

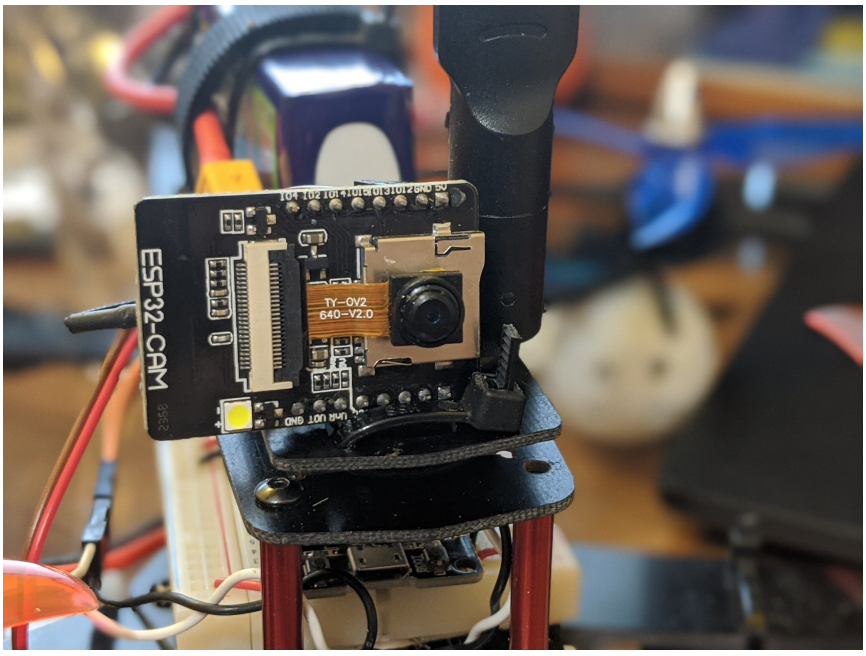


FIGURE 9.5: ESP32-CAM on the drone

Furthermore, the ESP32-CAM module can be programmed to detect specific events

or triggers using image processing algorithms. This allows the system to autonomously identify predefined objects, movements, or anomalies within the captured video stream. The integration of such intelligent video analytics enhances the efficiency and effectiveness of the surveillance system by automatically flagging potential threats or abnormal behavior for further investigation.

The recorded video footage from the ESP32-CAM module serves as a valuable resource for post-incident analysis, enabling in-depth review and evidence collection. The high-resolution video can be crucial in identifying individuals, vehicles, or other relevant details that might aid in subsequent investigations or legal proceedings.

By leveraging the capabilities of the ESP32-CAM module, the web-controlled drone becomes a comprehensive surveillance solution. The combination of live video streaming, remote camera control, and intelligent video analytics empowers authorities with enhanced monitoring capabilities, enabling them to proactively address security concerns and maintain a vigilant border surveillance system.

It is important to emphasize that privacy regulations and ethical considerations must be strictly adhered to when implementing video surveillance capabilities. Respect for privacy rights and compliance with applicable laws are essential to ensure the responsible and lawful use of the captured video footage.

Chapter 10

Start-up Quantum Solutions

10.1 How it began

Quantum Solutions traces its roots back to an inspiring academic journey that commenced as a Masters project at the University of Djilali Liabes in Sidi Bel Abbes, Algeria. Founded by four driven students, Hadjazi Mohammed Hisham, Sahraoui Mohammed Taher Amine, Mletta Mohammed Moncif, and Djebbar Yahya, and guided by the expertise of two esteemed professors, Pr. Boucli-Hacene Sofiane and Pr. Yousftae Abderhmane, our startup emerged from the Department of Computer Science, specializing in the field of Networking, Information Systems, and Security (RSSI).

The vibrant academic environment of Sidi Bel Abbes served as the cradle of our entrepreneurial journey. As we pursued our Masters in RSSI, we embarked on a challenging graduation project that aimed to address real-world security challenges.

Under the guidance and mentorship of Pr. Boucli-Hacene Sofiane and Pr. Yousftae Abderhmane, we delved into the realm of cutting-edge research, exploring the latest advancements in security technologies. Equipped with the knowledge and skills acquired during our rigorous academic curriculum, we harnessed our collective ambition to make a meaningful impact in the security domain.

Passionate about innovation and armed with a deep understanding of cryptography, wireless sensor networks, surveillance systems, and emerging post-quantum encryption algorithms, our team collaborated relentlessly. We dedicated countless hours to research, experimentation, and collaborative brainstorming sessions, refining our understanding of the security landscape and identifying opportunities for disruptive solutions.

The confluence of our academic training, unwavering determination, and the guidance provided by Pr. Boucli-Hacene Sofiane and Pr. Yousftae Abderhmane laid the foundation for Quantum Solutions. What began as a Masters project quickly evolved into a visionary startup, fueled by a clear mission to revolutionize the security industry.

Drawing upon the expertise gained at the University of Djilali Liabes and the Department of Computer Science, we set out to build Quantum Solutions—an entity driven by future technologies, quality assurance, and customer-centricity.

Today, Quantum Solutions proudly stands as a testament to the power of academic collaboration and entrepreneurial spirit. We owe our journey to the indomitable spirit of Hadjazi Mohammed Hisham, Sahraoui Mohammed Taher Amine, Mletta Mohammed Moncif, and Djebbar Yahya, whose dedication and perseverance led us forward.

The names of Pr. Boucli-Hacene Sofiane and Pr. Yousftae Abderhmane remain etched in our story, as they continue to inspire us with their guidance and shape our path.

The legacy of Quantum Solutions, born in Sidi Bel Abbes, Algeria, reminds us of the transformative power of education and the boundless potential of academic ventures. We are immensely grateful for the support and mentorship we received from the University of Djilali Liabes and the Department of Computer Science.

As we forge ahead, Quantum Solutions remains committed to pushing the boundaries of security innovation. We leverage our academic foundation to create cutting-edge solutions that empower individuals, businesses, industries, and governments to navigate the complex security landscape with confidence and peace of mind. The names of Hadjazi Mohammed Hisham, Sahraoui Mohammed Taher Amine, Mletta Mohammed Moncif, Djebbar Yahya, Pr. Boucli-Hacene Sofiane, and Pr. Yousftae Abderhmane will forever be intertwined with the success of Quantum Solutions, inspiring future generations to embark on their own transformative journeys.

10.2 The start

FIGURE 10.1: Quantum Solutions logo

Quantum Solutions is a dynamic start-up at the forefront of the security industry, specializing in providing advanced security solutions to individuals, businesses, industries, and governments. We are committed to revolutionizing the security landscape by leveraging future technologies and a relentless focus on quality assurance.

Our comprehensive range of security products and services caters to diverse security needs. For individuals, we offer state-of-the-art security kits that empower homeowners to protect their families, homes, and personal belongings. Our kits incorporate cutting-edge technologies, including smart alarm systems, intelligent surveillance cameras, and user-friendly mobile applications for seamless control and monitoring.

For businesses, industries, and governments, we deliver innovative security installations that leverage wireless sensor network (WSN) nodes, high-quality cameras with drone capabilities, and post-quantum proof encryption systems. These advanced installations provide robust security solutions, ensuring the protection of critical assets, infrastructures, and borders. We specialize in border control and monitoring, offering enhanced detection capabilities, rapid response mechanisms, and resilient security infrastructure.

At Quantum Solutions, we are driven by a passion for future technologies. We invest heavily in research and development to identify and integrate emerging technologies such as artificial intelligence, machine learning, and quantum-resistant encryption into our security solutions. This enables us to stay ahead of potential threats and provide our customers with state-of-the-art security measures that are future-proof.

Quality is the cornerstone of our operations. We have implemented stringent quality assurance measures throughout our product development, installation, and maintenance processes. Our dedicated team of experts ensures that all our security kits and installations undergo rigorous testing, adhering to the highest industry standards. We are committed to delivering reliable, durable, and effective security solutions that instill trust and confidence in our customers.

At Quantum Solutions, we believe in the power of collaboration. We actively seek partnerships with industry leaders, research institutions, and government agencies to drive advancements in security technologies. By participating in industry discussions, conferences, and initiatives, we aim to shape the future of security while advocating for the adoption of best practices and standards.

As a company, we are dedicated to creating a safer world. Our mission is to provide individuals, businesses, industries, and governments with the highest level of security available, ensuring safety, privacy, and peace of mind. Join us on this exciting journey as we redefine the boundaries of security and empower our customers to face the challenges of tomorrow with confidence. Quantum Solutions—where innovation meets security.

Chapter 11

Operations and Production

In Sidi bel abbes, Algeria, Quantum Solutions maintains its operations office, overseeing various aspects of the business. The operations team is responsible for managing day-to-day activities, including sales, marketing, customer support, and administrative functions. This local presence ensures efficient coordination with the manufacturing partner, logistics providers, and other stakeholders.

We employ multiple sales channels to reach a diverse customer base. Firstly, the company operates an online sales platform through its website, offering customers the convenience of purchasing security kits directly. Additionally, Quantum Solutions establishes partnerships with retail outlets to make its products accessible to consumers through physical stores. This omni-channel approach maximizes market reach and caters to different customer preferences.

focusing on establishing long-term contracts with government agencies and large enterprises. By forging strategic partnerships with these entities, the company can provide its advanced security solutions to safeguard critical infrastructure and confidential data. These contracts provide stability and contribute to the company's revenue growth and reputation.

11.1 Outsourcing strategy for production and packaging

Quantum Solutions has strategically chosen to outsource the production and packaging of its security kits to a reputable manufacturing partner in China. This decision enables the company to leverage the cost-effectiveness and expertise available in the Chinese market. By partnering with a reliable manufacturer, Quantum Solutions ensures high-quality production while focusing on core business activities.

We collaborates closely with its manufacturing partner to design secure and visually appealing packaging for its security kits. This packaging ensures the safe transportation and storage of the products. Logistics operations are carefully coordinated to facilitate timely delivery of finished products from China to various destinations, including the company's operations office in Sidi bel abbes, Algeria.

11.2 Quality control measures and regulatory compliance

Maintaining stringent quality control measures is paramount for Quantum Solutions. The company works closely with its manufacturing partner to enforce rigorous quality standards throughout the production process. Regular inspections, product testing, and certifications are conducted to guarantee that the security kits meet the highest levels of quality, functionality, and compliance with industry regulations.

Here Quantum Solutions is committed to continuous improvement across its operations and production processes. The company regularly evaluates its performance, seeking feedback from customers, partners, and employees to identify areas for enhancement. By staying updated with the latest industry advancements, Quantum Solutions ensures it remains at the forefront of innovation and delivers the highest level of service to its customers.

11.2.1 Brand quality controll

At Quantum Solution, we understand the critical importance of quality control in the market. As a leading brand, we prioritize maintaining a strong reputation through consistent delivery of high-quality products. We have established stringent quality standards and guidelines that govern every aspect of our product development, manufacturing, and deployment processes. Our commitment to quality is upheld through robust quality assurance procedures, including comprehensive testing, inspections, and validations throughout the product lifecycle.

To ensure the highest level of quality, we carefully select and manage our suppliers, conducting rigorous evaluations and audits to maintain strict quality control. We continuously strive for improvement, actively seeking customer feedback and monitoring product performance to identify areas for enhancement. By embracing quality improvement initiatives such as Lean and Total Quality Management (TQM), we drive ongoing improvements in our products and processes.

Compliance with industry standards, regulations, and certifications is of utmost importance to us. We take great care to ensure our products adhere to the highest standards, providing our customers with confidence and peace of mind. In the event of any product issues or recalls, our efficient recall management processes and dedicated customer support team ensure swift resolution and customer satisfaction.

We recognize that quality control directly impacts our brand reputation. By consistently delivering products that meet and exceed customer expectations, we enhance our reputation for reliability and trustworthiness in the competitive Security and survailnce market. Our focus on quality control is at the core of our commitment to providing cutting-edge, secure, and reliable solutions.

At Quantum Solution, quality control is not just a process; it is an integral part of our brand identity and customer-centric approach. We are dedicated to maintaining the highest standards of quality throughout every aspect of our business, ensuring that our customers have the utmost confidence in our products. Together, we can shape a secure and connected future in the market.

11.2.2 Customer quality controlle

At Quantum Solution, we place our customers at the heart of everything we do, and our commitment to quality control in the Security and survailnce market reflects that. We understand that our customers rely on our products for their critical needs, and we prioritize delivering the highest quality solutions to meet and exceed their expectations. Quality control is embedded in our DNA, as we adhere to stringent quality standards and guidelines throughout our product development, manufacturing, and deployment processes. Our rigorous quality assurance procedures, including thorough testing, inspections, and validations, ensure that our products perform at their best, providing reliability and peace of mind to our customers.

We understand that our customers' success depends on the seamless integration and performance of our products. Therefore, we carefully select and manage our suppliers, guaranteeing that they meet our stringent quality requirements. By doing so, we ensure that our customers receive products that meet the highest quality standards in the industry. Continuous improvement is a fundamental part of our approach. We actively seek feedback from our customers and closely monitor product performance, allowing us to identify areas for enhancement. By continuously improving our products based on customer insights, we provide solutions that evolve alongside their needs, delivering the highest level of value.

We prioritize compliance with industry standards, regulations, and certifications to ensure that our customers can trust the security and reliability of our products. We understand that their trust is built on our commitment to maintaining the highest quality standards, and we take that responsibility seriously. In the rare event of any product issues or recalls, our dedicated customer support team is here to provide timely assistance and resolve any concerns. We value our customers' satisfaction above all else and strive to ensure that their experience with our products is exceptional. At Quantum Solution, our focus on quality control is driven by our dedication to our customers' success. We understand that by delivering high-quality solutions, we empower our customers to achieve their goals and meet their objectives in the ever evolving market. Together, we can build a strong and lasting partnership, where quality is the foundation of our relationship. We are committed to going above and beyond to meet our customers' needs, delivering the reliability, performance, and security they deserve in their Security and survailnce solutions.

11.2.3 Quality control on products

At Quantum Solution, we are dedicated to delivering high-quality products in the Security and survailnce market, driven by our commitment to excellence. Our focus on quality control ensures that our products stand out in terms of reliability, performance, and security, offering our customers a superior experience. Quality control is an integral part of our product development process. From conceptualization to manufacturing, we adhere to stringent quality standards and guidelines. Our products undergo rigorous testing, inspections, and validations at every stage to ensure optimal performance and durability. We understand that our customers rely on our products for critical applications, and we take that responsibility seriously. By employing advanced quality assurance procedures and implementing cutting-edge technologies, we deliver solutions that consistently meet and exceed customer expectations. We are relentless in our pursuit of continuous improvement. Customer feedback plays a vital role in our product enhancement efforts, as we actively seek input to refine and optimize our offerings. Through constant monitoring and evaluation, we stay at the forefront of technological advancements, providing our customers with innovative and future proof solutions.

Compliance with industry standards and certifications is of utmost importance to us. Our products undergo comprehensive testing to ensure they meet the highest industry benchmarks for security, interoperability, and performance. By adhering to these standards, we instill confidence in our customers, knowing that they are investing in products that meet the most stringent requirements. In the rare event of any product issues, our dedicated support team is readily available to address

customer concerns and provide timely resolutions. We understand that customer satisfaction is paramount, and we are committed to standing behind our products throughout their lifecycle. our focus on quality control is driven by our passion for delivering products that make a difference. We strive to empower our customers with reliable, high performance Security solutions that enable them to achieve their goals and overcome challenges. We are proud of the quality and craftsmanship that goes into each product we offer. our customers gain not just a product but a trusted partner committed to their success.

11.3 ISO standards

The ISO 9001:2015 certification process involved a rigorous evaluation of our quality management system by accredited certification bodies. These independent entities thoroughly assessed our organizational processes, procedures, and controls to determine their alignment with the stringent requirements of the ISO standard. We successfully underwent a stage 1 audit, which evaluated our readiness, followed by a comprehensive stage 2 audit to evaluate the effectiveness and efficiency of our quality management system.

FIGURE 11.1: ISO 9001:2015

Furthermore, as we operate in the realm of RF frequencies, we have taken into consideration the specific standards and regulations pertinent to our industry. In addition to ISO 9001:2015, we have ensured compliance with ISO/IEC 17025, which focuses on the competence and impartiality of calibration and testing laboratories. We have also adhered to industry-specific standards such as EN 301 489 for electromagnetic compatibility (EMC) and EN 300 220 for radio frequency emissions, ensuring that our products and services meet the highest standards of performance, safety, and compliance.

FIGURE 11.2: ISO/IEC 17025

We remain vigilant in staying updated with the latest versions and requirements of ISO standards and industry-specific regulations to ensure ongoing compliance and alignment with best practices. Our ISO certification provides us with a solid foundation to continually enhance our processes, drive innovation, and consistently meet and exceed the expectations of our valued customers.

11.3.1 ISO for MicroWave frequencies

At Quantum Solution, we recognize the significance of ISO standards in ensuring the highest quality and performance of our products and services, particularly in the field of Microwave frequencies. ISO, which stands for the International Organization for Standardization, sets internationally recognized standards that guide various industries, including Microwave technologies. ISO has developed specific standards for Microwave frequencies to ensure that companies operating in this domain adhere to best practices and deliver products and services that meet rigorous quality criteria. One such standard is ISO/IEC 17025, which specifically addresses the competence and impartiality of testing and calibration laboratories in the field of Microwave frequencies. ISO/IEC 17025 provides guidelines for the establishment and operation of laboratories, encompassing their management, personnel competence, testing procedures, calibration methods, equipment, and quality assurance processes. By adhering to this standard, we demonstrate our commitment to maintaining a high level of accuracy, reliability, and consistency in the measurements and tests related to Microwave frequencies.

FIGURE 11.3: ISO/IEC 17025

In addition to ISO/IEC 17025, there are other ISO standards relevant to Microwave technologies. ISO/IEC 90003 focuses on software engineering in the Microwave domain, providing guidelines for the development, implementation, and maintenance of software used in Microwave applications. This standard ensures that our software solutions meet quality standards and comply with industry best practices.

FIGURE 11.4: ISO/IEC 90003

ISO/IEC 18000 is another important standard related to Microwave frequencies, specifically addressing Microwave identification and tracking technologies. This standard sets guidelines for the design, implementation, and performance of Microwave identification systems, such as RFID (Radio Frequency Identification) technologies. By adhering to ISO/IEC 18000, we ensure that our Microwave identification and tracking solutions meet international standards and deliver accurate and reliable results. By adhering to these ISO standards for Microwave frequencies, we demonstrate our commitment to quality, precision, and customer satisfaction. These standards serve as a framework for our quality management system, ensuring that our products and services meet or exceed industry benchmarks. They also provide assurance to our customers that our operations align with international best practices and that our solutions are designed and implemented with the utmost attention to quality and performance. At Quantum Solution, we continuously strive to stay up-to-date with the latest developments in ISO standards for Microwave frequencies. We invest in ongoing training and development for our team, ensuring they possess the necessary knowledge and skills to implement these standards effectively. This allows us to provide cutting- edge Microwave solutions that not only meet ISO requirements but also address the specific needs and challenges of our customers. In summary, ISO standards for Microwave frequencies are an essential component of our quality management system at Quantum Solution. By adhering to these standards, we assure our customers that our products and services meet the highest industry benchmarks for accuracy, reliability, and performance in the realm of Microwave technologies. Our commitment to ISO standards underscores our dedication to excellence, customer satisfaction, and the delivery of innovative Microwave solutions that empower our customers to succeed.

11.3.2 ISO for our plastics

At Quantum Solution we place great emphasis on adhering to ISO standards to ensure the quality and reliability of our products, particularly in relation to plastics used in cases. ISO, which stands for the International Organization for Standardization, develops internationally recognized standards that guide various industries, including the use of plastics in case manufacturing. ISO has established specific standards for plastics used in cases to ensure the materials meet stringent quality requirements and contribute to the overall performance and durability of our products. One such standard is ISO 9001:2015, which sets guidelines for a quality management system that encompasses the design, development, production, and delivery of products. By complying with ISO 9001:2015, we demonstrate our commitment to consistently meeting customer expectations and delivering products that meet the highest quality standards.

FIGURE 11.5: ISO 9001:2015

In addition to ISO 9001:2015, there are other ISO standards relevant to plastics used in cases. ISO 19001 provides guidelines for the audit of a quality management system, ensuring that our manufacturing processes are regularly reviewed and evaluated to maintain the highest level of quality. This standard helps us identify areas for improvement and implement corrective actions, leading to continuous enhancement of our case manufacturing practices. ISO 23900-1 focuses specifically on the environmental aspects of plastics used in electrical and electronic equipment. This standard guides us in selecting materials that are environmentally friendly, promote sustainability, and comply with regulations related to the use and disposal of plastics. By adhering to ISO 23900-1, we ensure that our cases not only meet performance requirements but also align with environmental standards.

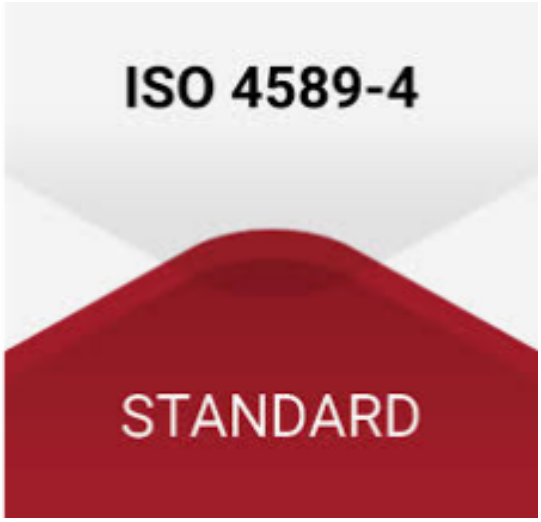


FIGURE 11.6: ISO 4589-4

ISO 4589-4 is another significant standard that relates to the fire performance of plastics used in cases. It specifies the testing methods and performance criteria to assess the flammability and fire resistance of plastics. By following ISO 4589-4, we ensure that our cases provide appropriate fire protection, reducing the risk of damage or harm in the event of a fire. By adhering to these ISO standards for plastics used in cases, we demonstrate our commitment to producing high-quality, durable, and environmentally responsible products. These standards serve as a foundation for our quality management system, driving our continuous improvement efforts and ensuring that our cases meet or exceed industry standards. At Quantum Solution , we prioritize staying up-to-date with the latest developments in ISO standards for plastics. Our team undergoes regular training and education to ensure they have the necessary knowledge and skills to implement these standards effectively. This enables us to manufacture cases that not only comply with ISO requirements but also meet the specific needs and expectations of our customers. In summary, ISO standards for plastics used in cases are integral to our quality management system at Quantum Solution . By adhering to these standards, we guarantee that our products meet the highest industry benchmarks for quality, environmental responsibility, and fire performance. Our commitment to ISO standards underscores our dedication to providing our customers with reliable, durable, and sustainable case solutions that protect their valuable equipment and meet their evolving needs.

11.3.3 ISO for our electronics

At Quantum Solution, we place great emphasis on adhering to ISO standards to ensure the quality and reliability of our electronic products. ISO, which stands for the International Organization for Standardization, develops internationally recognized standards that guide various industries, including the electronics sector. ISO has established specific standards for electronics to ensure that our products meet stringent quality requirements and comply with industry best practices. One key standard is ISO 9001:2015, which sets guidelines for a quality management system that encompasses the design, development, production, and delivery of electronic products. By complying with ISO 9001:2015, we demonstrate our commitment to consistently meeting customer expectations and delivering products that meet the

highest quality standards. In addition to ISO 9001:2015, there are other ISO standards relevant to electronics. ISO 13485 focuses specifically on the medical device industry, providing guidelines for the design, development, production, and distribution of medical electronic devices. This standard ensures that our medical electronic products meet the necessary regulatory requirements, including safety, reliability, and effectiveness. ISO 14001 is an environmental management standard that guides us in minimizing the environmental impact of our electronic products and manufacturing processes. By adhering to ISO 14001, we demonstrate our commitment to sustainable practices, waste reduction, and energy efficiency throughout the product lifecycle.

FIGURE 11.7: ISO 14001

ISO 27001 is another significant standard that relates to information security management. In an increasingly interconnected world, protecting sensitive electronic data is crucial. By following ISO 27001, we ensure that our information security practices, including data protection, risk management, and incident response, meet international standards and safeguard our customers' information. By adhering to these ISO standards for electronics, we demonstrate our commitment to producing high-quality, reliable, and sustainable electronic products. These standards serve as a foundation for our quality management system and environmental practices, driving our continuous improvement efforts and ensuring that our products meet or exceed industry standards. At Quantum Solution, we prioritize staying up-to-date with the latest developments in ISO standards for electronics. Our team undergoes regular training and education to ensure they have the necessary knowledge and skills to implement these standards effectively. This enables us to manufacture electronic products that not only comply with ISO requirements but also meet the specific needs and expectations of our customers. In summary, ISO standards for electronics are integral to our quality management system at Quantum Solution. By adhering to these standards, we guarantee that our electronic products meet the highest industry benchmarks for quality, regulatory compliance, environmental responsibility, and information security. Our commitment to ISO standards underscores our dedication to providing our customers with electronic solutions that are of the highest quality, reliable, and in line with global best practices.

11.3.4 ISO for our lithium batteries

At Quantum Solution, we recognize the importance of adhering to ISO standards to ensure the safety and performance of lithium batteries. ISO, which stands for the International Organization for Standardization, has developed specific standards for lithium batteries to address the unique characteristics and potential risks associated with these energy storage devices. ISO 12405 provides guidelines for the characterization, testing, and safety requirements of lithium-ion batteries. This standard covers various aspects such as electrical performance, thermal behavior, mechanical integrity, and environmental considerations. By following ISO 12405, we ensure that our lithium batteries meet the necessary safety requirements and perform reliably in a wide range of applications. ISO 15194 focuses specifically on electrically powered cycles, including e-bikes and pedelecs, which utilize lithium batteries. This standard outlines requirements for the design, construction, and performance of lithium battery systems in these vehicles, ensuring their safe and efficient operation. ISO 6469 provides safety requirements for battery electric vehicles (BEVs) that utilize lithium batteries. This standard addresses aspects such as electrical safety, thermal management, protection against external influences, and emergency response procedures. By adhering to ISO 6469, we ensure that our lithium battery systems meet the necessary safety standards for use in electric vehicles. Furthermore, ISO 27000 series, specifically ISO 27001, is relevant to the information security management of lithium battery systems. This standard helps us establish and maintain a robust information security management system, safeguarding critical data and ensuring the privacy and integrity of information associated with our lithium batteries.

FIGURE 11.8: ISO 27000

By adhering to these ISO standards for lithium batteries, we demonstrate our commitment to manufacturing batteries that meet or exceed the highest safety, performance, and security standards. We prioritize the implementation of rigorous testing, quality control processes, and safety measures throughout the battery production and supply chain. AtQuantum Solution, we continually invest in research and development to stay at the forefront of lithium battery technology and advancements. Our team of experts closely monitors updates to ISO standards, ensuring that our lithium batteries are always compliant with the latest guidelines and industry best practices. In summary, ISO standards for lithium batteries form an integral part of our commitment to producing safe, reliable, and high-performance energy storage

solutions at Quantum solution. By adhering to these standards, we ensure that our lithium batteries meet the necessary safety and quality requirements, providing our customers with batteries they can trust for a wide range of applications. Our dedication to ISO standards underscores our focus on delivering lithium battery solutions that prioritize safety, performance, and customer satisfaction.

11.4 Supply chain and logistics

Efficient supply chain management is crucial to ensure the smooth flow of materials and components required for the production of security kits. Quantum Solutions establishes strong relationships with trusted suppliers who demonstrate reliability, consistent quality, and adherence to agreed-upon delivery schedules. Implementing effective inventory management practices minimizes stockouts and optimizes inventory levels.

Chapter 12

Conclusion

In conclusion, this thesis represents a significant contribution to the field of Wireless Sensor Networks (WSNs) by addressing critical challenges related to secure and reliable communication. The integration of post-quantum cryptography and self-healing mechanisms has resulted in a comprehensive solution that ensures the confidentiality, integrity, and availability of data in WSNs. By designing and analyzing a post-quantum cryptographic scheme suitable for WSNs, the thesis has provided a robust defense against both classic and quantum attacks. Furthermore, the incorporation of a self-healing mechanism has empowered the network to autonomously recover from node failures and attacks, enhancing network resilience and effectiveness.

The patented prototype, "Border surveillance and monitoring using solar-powered Wireless Sensor Network equipped with 24GHz mmWave sensors and GPS Modules for tracking and ESP-NOW and LoRa protocols for communication," has exemplified the practical application of the proposed combined scheme in the domain of border surveillance and monitoring. This innovative prototype showcases the utilization of cutting-edge technologies such as solar power, high-frequency mmWave sensors, precise GPS tracking, and efficient communication protocols. Through its integration, the prototype offers enhanced security, environmental sustainability, and exceptional performance.

Beyond theoretical contributions, the establishment of Quantum Solutions as a start-up adds another dimension to the impact of this research. With a strong focus on delivering comprehensive security services and innovative security kits, Quantum Solutions aims to address the diverse needs of households, small businesses, and large-scale government security projects. Leveraging the patented prototype and pending patents, the start-up is well-positioned to offer state-of-the-art security solutions that combine reliability, effectiveness, and innovation.

Moving forward, there are several promising avenues for future research and development. Firstly, further exploration of post-quantum cryptographic schemes tailored to different WSN applications can enhance security and resilience against emerging threats. By investigating advanced self-healing mechanisms such as fault-tolerant protocols and adaptive recovery strategies, the robustness and efficiency of WSNs can be further improved, particularly in dynamic and unpredictable environments. Moreover, continuous monitoring and evaluation of the practical implementation of the proposed scheme within Quantum Solutions will provide valuable insights for

refining and optimizing the technologies based on real-world feedback.

In summary, this thesis has demonstrated the successful integration of post-quantum cryptography and self-healing mechanisms in WSNs, resulting in a comprehensive solution for secure and robust communication. The patented prototype, coupled with the establishment of Quantum Solutions, further reinforces the practical relevance and market potential of this research. With a strong foundation in theoretical advancements and practical applications, this work contributes to the advancement of secure WSN communication, with the potential to shape the future of the security industry.

Bibliography

[Asi21] Rameez Asif. "Post-Quantum Cryptosystems for Internet-of-Things: A Survey on Lattice-Based Algorithms". en. In: *IoT* 2.1 (Feb. 2021), pp. 71–91. ISSN: 2624-831X. DOI: 10.3390/iot2010005. URL: https://www.mdpi.com/2624-831X/2/1/5 (visited on 02/13/2023).

[blo23] blog.expo.dev. *Exposition*. July 2023. URL: https://blog.expo.dev/ (visited on 07/06/2023).

[BS] Themistoklis Bourdenas and Morris Sloman. "Self-healing in Wireless Sensor Networks". PhD thesis. London, UK: Imperial College , Department of Computing. URL: http://www.doc.ic.ac.uk/%E2%88%BCtbourden.

[Che+17] Chi Cheng et al. "Securing the Internet of Things in a Quantum World". In: *IEEE Communications Magazine* 55.2 (Feb. 2017), pp. 116–120. ISSN: 0163-6804. DOI: 10.1109/MCOM.2017.1600522CM. URL: http://ieeexplore.ieee.org/document/7842421/ (visited on 02/13/2023).

[cod] code.visualstudio.com. *Visual Studio Code - Code Editing. Redefined*. en. URL: https://code.visualstudio.com/ (visited on 07/06/2023).

[doc] docs.expo.dev. *Expo Documentation*. en. URL: https://docs.expo.dev/ (visited on 07/06/2023).

[doc23] docs.reactnative.dev. *Introduction · React Native*. en. June 2023. URL: https://reactnative.dev/docs/getting-started (visited on 07/06/2023).

[Fer20] Tiago M. Fernandez-Carames. "From Pre-Quantum to Post-Quantum IoT Security: A Survey on Quantum-Resistant Cryptosystems for the Internet of Things". In: *IEEE Internet of Things Journal* 7.7 (July 2020), pp. 6457–6480. ISSN: 2327-4662, 2372-2541. DOI: 10.1109/JIOT.2019.2958788. URL: https://ieeexplore.ieee.org/document/8932459/ (visited on 02/13/2023).

[Guo+18] Hua Guo et al. "Self-healing group key distribution protocol in wireless sensor networks for secure IoT communications". en. In: *Future Generation Computer Systems* 89 (Dec. 2018), pp. 713–721. ISSN: 0167739X. DOI: 10.1016/j.future.2018.07.009. URL: https://linkinghub.elsevier.com/retrieve/pii/S0167739X18304102 (visited on 02/13/2023).

[int22] intellectsoft.net. *Mobile App Architecture: Best Practices in 2023 | Intellectsoft*. en-US. Dec. 2022. URL: https://www.intellectsoft.net/blog/mobile-app-architecture/ (visited on 07/06/2023).

[JIA+17] LIU JIANYUAN et al. "WSN MONITORING SYSTEM". CN206629277. Oct. 2017.

[KA] Mrunal Mohan Khedkar and Gajendra Mahadeorao Asutkar. "INTRUDER MOTION DETECTION SYSTEM". IN201921024557.

[ldp] ldplayer.net. *LDPlayer - Émulateur Android léger et rapide pour PC*. fr. URL: https://fr.ldplayer.net (visited on 07/06/2023).

[MO15] Atsuko Miyaji and Kazumasa Omote. "Self-healing wireless sensor networks". en. In: *Concurrency and Computation: Practice and Experience* 27.10

(July 2015), pp. 2547–2568. ISSN: 1532-0626, 1532-0634. DOI: 10.1002/cpe.3434. URL: https://onlinelibrary.wiley.com/doi/10.1002/cpe.3434 (visited on 02/13/2023).

[MS16] Mohamed-Lamine Messai and Hamida Seba. "A survey of key management schemes in multi-phase wireless sensor networks". en. In: *Computer Networks* 105 (Aug. 2016), pp. 60–74. ISSN: 13891286. DOI: 10.1016/j.comnet.2016.05.005. URL: https://linkinghub.elsevier.com/retrieve/pii/S1389128616301360 (visited on 02/13/2023).

[Pat+] Pravin P Patil et al. "SYSTEM FOR CONTROLLING AND MONITORING SOLAR-POWERED, WIRELESS SENSOR NODES IN PRECISION AGRICULTURE". IN202211052636.

[PK22] Neeraj Patel and Vinod Kumar. "An efficient key distribution Scheme for WSN with Mutual Healing Capability". en. In: *Multimedia Tools and Applications* 81.25 (Oct. 2022), pp. 36735–36749. ISSN: 1380-7501, 1573-7721. DOI: 10.1007/s11042-022-13501-y. URL: https://link.springer.com/10.1007/s11042-022-13501-y (visited on 02/13/2023).

[Pre18] John Preskill. "Quantum Computing in the NISQ era and beyond". en. In: *Quantum* 2 (Aug. 2018), p. 79. ISSN: 2521-327X. DOI: 10.22331/q-2018-08-06-79. URL: https://quantum-journal.org/papers/q-2018-08-06-79/ (visited on 02/13/2023).

[rea] reactnative.dev. *React Native · Learn once, write anywhere*. en. URL: https://reactnative.dev/ (visited on 07/06/2023).

[RP13] Tomasz Rams and Piotr Pacyna. "A Survey of Group Key Distribution Schemes With Self-Healing Property". In: *IEEE Communications Surveys & Tutorials* 15.2 (2013), pp. 820–842. ISSN: 1553-877X. DOI: 10.1109/SURV.2012.081712.00144. URL: http://ieeexplore.ieee.org/document/6329366/ (visited on 02/13/2023).

[Tia+11] B. Tian et al. "Self-Healing Key Distribution Schemes for Wireless Networks: A Survey". en. In: *The Computer Journal* 54.4 (Apr. 2011), pp. 549–569. ISSN: 0010-4620, 1460-2067. DOI: 10.1093/comjnl/bxr022. URL: https://academic.oup.com/comjnl/article-lookup/doi/10.1093/comjnl/bxr022 (visited on 02/13/2023).

[VV19] China Venkaiah Vadlamudi and Sesha Phani Deepika Vadlamudi. "A novel self-healing key distribution scheme based on vector space access structure and MDS codes: Self-healing Key distribution using MDS codes". en. In: *International Journal of Communication Systems* 32.16 (Nov. 2019), e4088. ISSN: 10745351. DOI: 10.1002/dac.4088. URL: https://onlinelibrary.wiley.com/doi/10.1002/dac.4088 (visited on 02/13/2023).

[Wan11] Qiuhua Wang. "Practicality analysis of the self-healing group key distribution schemes for resource-constricted wireless sensor networks". In: *2011 Third International Conference on Communications and Mobile Computing* (2011). DOI: 10.1109/cmc.2011.113.

[WM21] Vaz Wilfred J and Sarvagya Mrinal. "DESIGN AND IMPLEMENTATION OF SCALABLE WIRELESS SENSOR NETWORK BEYOND WIFI". English. IN202141013947. Mar. 2021.