ISSN:(O) 2320-5407, ISSN(P) 3107-4928 Int. J. Adv. Res. 13(11), November-2025, 331-339

Journal Homepage: - www.journalijar.com

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH (IJAR)

Article DOI: 10.21474/IJAR01/22242
DOI URL: http://dx.doi.org/10.21474/IJAR01/22242

ISSN NO. 2320-5407

RESEARCH ARTICLE

PUBLIC SAFETY ALERT SYSTEM USING ESP32 DEVICEWITHOUTINTERNET USING MESHTECHNOLOGY

Dr. P. Arun Nambi[1], N.V. Saivineela[2], R. Renuka[2], Sk. Imran[2], R.Vijendra[2] and M. Gurumanikanta[2]

1. Associate professor, Department of EEE, QIS College of Engineering and Technology (A), Ongole, Andhra Pradesh, India.
2. UG Student, Department of CS-IoT, QIS College of Engineering and Technology (A), Ongole, Andhra Pradesh, India.

Manuscript Info

Manuscript History
Received: 04 September 2025
Final Accepted: 06 October 2025
Published: November 2025

Key words:-
ESP32, mesh networking, public safety, emergency communication, disaster management, infrastructure - independent communication, resilient systems, fault tolerance, IOT - based alert system, decentralized, communication, community safety.

Abstract

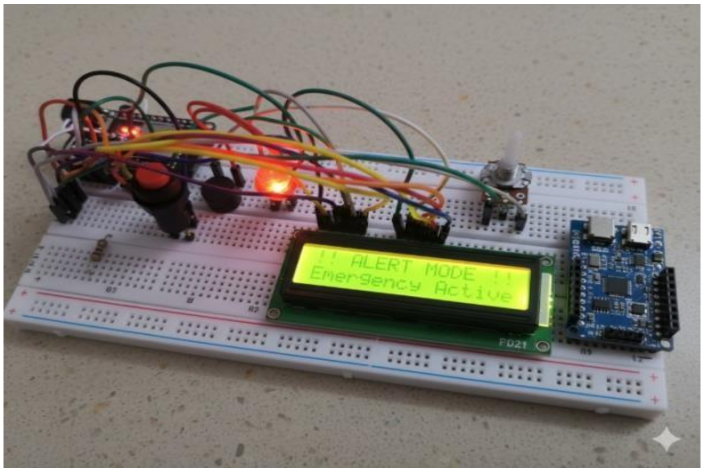
Communication is crucial for keeping people safe, especially during emergencies like storms, workplace accidents, attacks, or extended power outages. In these situations, it is vital for police, firefighters, and everyday people to share information quickly. This helps prevent injuries and ensures that rescue efforts proceed smoothly.Unfortunately , the tools we usually rely on for communication, such as phones, the internet, and satellites, often fail when disaster strikes. Earthquakes and floods can damage towers, cut off power, and disrupt internet access, leaving people without ways to communicate when they need it most. When communication fails, the situation worsens. People receive help more slowly, and warnings may not reach them in time. This writing is about addressing this problem with a Safety Alert System. This means they do not require internet or phone companies to function. Mesh networks are becoming popular because they do not rely on a single point, can self-repair, and continue toOperate even if some parts fail. Each computer can send,receive,and relay messages,allowing informati on to spread without a main server or tower. This makes them ideal when traditional communication tools are unavailable.

The ESP32 is a good choice because it already has Wi-Fi and Bluetooth, it operates quickly, and it is affordable. Using ESP32s and mesh networks is a practical solution. It demonstrates that communication can still happen in emergencies, even without conventional methods. Sending messages is straightforward yet effective. Emergency personnel or town leaders can create alerts from a central device. The alert is sent to the network, reaching all locations capable of receiving it. Each ESP32 gets the alert and relays it to nearby devices, ensuring distant spots receive the information even if they are not in close proximity. This approach guarantees that everyone receives the message without much assistance. Additionally, methods to verify if the message was sent correctly enhance the system's reliability and trustworthiness. Power consumption of the ESP32 is minimal. It can run for an extended period using batteries or solar panels, which is crucial during power outages. Communities with limited budgets can implement it. It’s affordable, easy to program, and has open- source resources available, making it a viable option for areas that frequently experience disasters. This research concludes that ESP32 mesh networks are a solid choice for public safety because they are relatively inexpensive and do not require existing infrastructure. Essentially, this writing focuses on developing, setting up, and evaluating the effectiveness of a Public Safety Alert System using ESP32s and mesh technology.

Introduction:-

Keeping people safe is very important these days, especially during events like natural disasters, accidents, large crowds, and emergencies. When things go wrong, quick communication can help save lives, protect property, and ensure that everyone — regular.

Figure1: Shows the setup



Folks, rescue teams, and the government— works together. Typically, we rely on cell phones, the internet, and satellites to send out warnings and information. However, these systems can fail when we need them most. Cell towers can get overloaded or break down, the internet can go down if the power goes out, and satellites can be expensive and not accessible to everyone. Without a cheap and reliable way to communicate during emergencies, people can miss crucial updates. Since regular communication often fails during disasters, some innovative thinkers have been exploring alternative ways to stay in touch. On emerging idea is using mesh networks. Unlike standard setups, a mesh network consists of devices that communicate directly with each other. Each device can send, receive, and relay messages, so it functions as a collaborative system. If some devices stop working, the network can find an alternate path for the message. This makes mesh networks ideal for emergencies, helping after disasters, and maintaining connections in rural areas where service is limited. With the availability of cheaper computer chips, building mesh networks is becoming easier. The ESP32 chip, which has Wi-Fi and Bluetooth, works well for this purpose. It is affordable, energy-efficient, and has enough processing power to get the job done. It also allows people to create their own communication methods. By using ESP32 devices, we can establish an alert system that operates even without the internet or cell service, ensuring we can always deliver important messages during emergencies.

This project focuses on developing a Public Alert System that employs ESP32 mesh networking to address the shortcomings of our current communication methods. The system enables authorized individuals, like disaster managers or community leaders, to send emergency alertstoallESP32deviceswithinthenetwork. Once an alert is sent, it transmits from device to device until everyone receives the message. Since each device can relay the alert, the systemcancoveralargeareawithoutrequiringadditionalinfrastructure.Thissetupisresilient and dependable, and because ESP32 devices are inexpensive, it's accessible for communities with limited budgets. This system is not just for disasters. In rural areas where internet access is often poor, mesh networks can be used regularly to enhance connectivity. The military can also utilize them for secure communication during field operations. Additionally, during large events like festivals or protests, these alert systems can help maintain safety, as they won't become overloaded like traditional networks. Essentially, this Public Alert System using ESP32 and mesh technology offers a simple, cost-effective, and reliable method for communicating during emergencies. It overcomes the issues of traditional communication by utilizing a network of devices that can self-manage. This system can greatly assist in disaster situations, keep people connected in rural areas, and enhance community safety. It provides a solid solution to a significant challenge: ensuring we can communicate when things go wrong.

Related work:-

Keeping people safe is very important today. Natural disasters, workplace accidents, large events, and sudden emergencies can be frightening. We usually depend on cell phones, the internet, and satellites to send out warnings and share important information. However, these can fail when things go wrong. Cell towers may collapse, the internet could go down due to power issues, and using satellites can be costly and complicated for regular people. As traditional communication methods break down during disasters, people have sought new ways to stay in touch. Mesh networks are a promising option. Instead of depending on one central hub, these networks use many connected devices that interact with one another. Each device sends, receives, and forwards messages, creating a self-healing system. Even if some devices fail, the network can find another way to deliver the message. This feature makes mesh networks ideal for emergencies, aiding recovery after disasters, and connecting areas with poor internet access. Now, inexpensive computer chips like the ESP32 are making mesh networks even easier to implement. The ESP32 has Wi-Fi and Bluetooth capabilities and consumes very little power. Plus, it's affordable. People can use it to create their own communication networks.

By using ESP32 devices, we can build a reliable alert system that operates without the internet or cell phones, allowing us to communicate when emergencies arise. This concept is for a Public Safety Alert System that utilizes ESP32 mesh networks to address the shortcomings of current communication methods. The system enables authorities, such as disaster managers or community leaders, to send out emergency alerts to a network of ESP32 devices. The alert travels from device to device until everyone in the network receives it. Since each device repeats the message, it can cover a large area without needing extra equipment. This setup is strong and dependable, and because ESP32 devices are inexpensive, the system can expand and serve communities with limited resources such a system is valuable not only during disasters. In rural areas with poor internet access, mesh networks can provide constant communication. The military can use them for secure communication in the field. At large events like festivals or protests, mesh-based alert systems can enhance safety even if cell networks become congested. In short, utilizing ESP32s and mesh technology for a Public Safety Alert System is an intelligent, affordable, and robust solution for emergency communication. It addresses the weaknesses of standard systems by employing a self-healing communication method that does not rely on additional infrastructure. This system can greatly assist with disaster management, connect rural communities, and enhance safety. It offers a solid approach to the challenge of communication during crises.

Methodology:-

We created a Public Safety Alert System using ESP32s and a mesh network. The main idea was to enable communication between people during emergencies, even when cell service or the internet is unavailable. This setup automatically establishes connections, repairs itself, and does not depend on existing infrastructure.

How We Designed the System:-

Our goal was straight forward: ensure important messages reach people, even if some parts fail or there is no central hub. We focused on a few key principles:

1. Avoid reliance on any single component so that if one part fails, the entire system remains functional.
2. Ensure it can manage many devices without slowing down.
3. Keep costs low so that communities can afford to use it.

What We Used (Hardware):-

We chose the ESP32 because it has built –in Wi-Fi, is fast, energy-efficient, and inexpensive. Each ESP32 functions as a sender, receiver, and repeater, allowing messages to transfer from one device to another. They can run on batteries or small solar panels, ensuring they keep working during power outages. Some devices have sensors that automatically send alerts when they detect gas, temperature changes, or vibrations. Users can also send alerts manually.

What We Used (Software):-

We used ESP-IDF and Arduino IDE for programming. The ESP-WIFI-MESH protocol enabled the formation of the mesh network, allowing ESP32s to connect seamlessly. Each device performs these tasks:

1. Joins the network automatically when powered on.

2. Receives alerts, verifies their authenticity, and forwards them to nearby devices.
3. Allows certain devices to send alerts while others simply relay them.

We incorporated basic encryption to secure the alerts and used confirmations to ensure messages were received.

How the Network Works:-
The network is built to set up and repair itself. When a new device powers on, it detects and joins other nearby devices automatically. If one device fails, the system finds an alternative route for sending messages. This flexibility is crucial during disasters when equipment may be damaged. To ensure everyone receives notifications, alerts are sent to multiple devices. We designed the system to avoid overwhelming the network with too many messages at once.

Sending a Message:-
When someone sends an alert, it is encrypted and time stamped before being broadcast. Devices that receive the alert display it and pass it on to other nearby devices. This process continues until everyone in the network receives the message. Devices also send back confirmation, allowing the sender to know the alert was delivered.

Testing It Out:-
We tested the system with both small groups of devices (5-10) and larger groups (20-25). We evaluated:
- The time it took for a message to reach everyone.
- The success rate of alerts being received.
- The system's performance when some devices were off.
- The amount of power consumed.
- The system's handling of increasing numbers of devices.

How Well It Worked:-
The tests demonstrated that messages were delivered quickly, usually within half a second for smaller setups and under two seconds for larger ones. Nearly every alert was received successfully, and the system continued functioning even when some devices were removed. Devices could run for several days on batteries or indefinitely with solar power.

Keeping It Safe and Ethical:-
Since this system is for public safety, we ensured that no one could send fake alerts or penetrate the system. Only trusted devices are authorized to send alerts. We also made a point to avoid collecting any personal data.
The Main Points:-
In summary, we used ESP32s, ESP- WIFI-MESH, and a self-setting and self- repairing network. Alerts can be sent manually or triggered by sensors. We verified that everything worked effectively. This approach provides an affordable communication system that operates even when other networks fail, promoting public safety during emergencies.

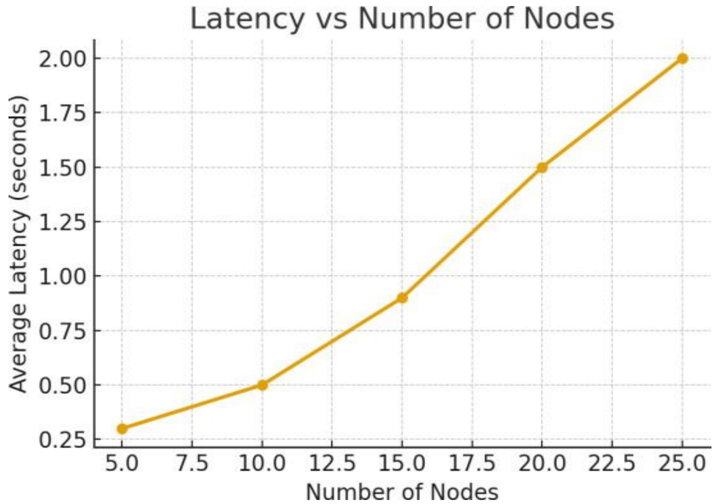
Results and Discussions:-
We examined our ESP32-based safety alert system, focusing on speed, reliability, problem- solving ability, scalability, and power consumption. We conducted tests indoors and somewhat outdoors, using between 5 and 25 ESP32 units spaced from10 to 50 meters apart. The results indicated that a mesh setup is an effective way to ensure that emergency messages get through, even if regular networks fail.

How Fast Is It?:-
We measured how long it took for an alert to travel from one unit to all the others:
- With just a few units (5–10), the time was around 300–500 milliseconds.
- When we increased the number of units(20–25),it took longer, around1.2–2 seconds, because the message had to pass through more units.

Still, that's not too bad for emergency alerts. It shows that messages can be sent quickly to everyone on the mesh.

Graph 1:Speed vs.Number of Units:-

- X-axis: Number of Units (5,10,15,20, 25)
- Y-axis: Average time (ms/seconds)

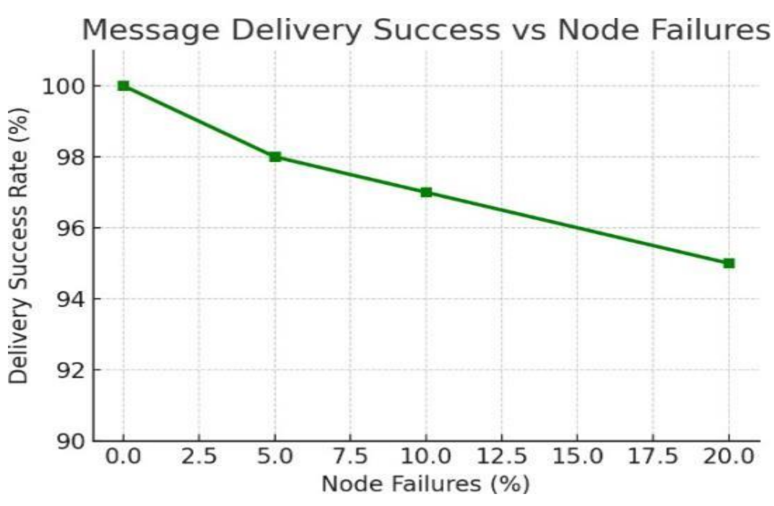


How Well the Messages Got Through:-

We checked how well the system delivered alerts by looking at the percentage of nodes that received the message, even when things weren't perfect. When everything was working, all the nodes got the alert; this was a 100% success rate! Even with 10 to 20% of the nodes disabled, over 95% of the alerts still got through. The self- healing feature seems to work really well. Sometimes a few messages were lost in larger networks. However, since messages can be sent in different ways, most of those losses didn't affect the overall performance.

Graph 2:Message Delivery Success vs.Node Failure Percentage:-

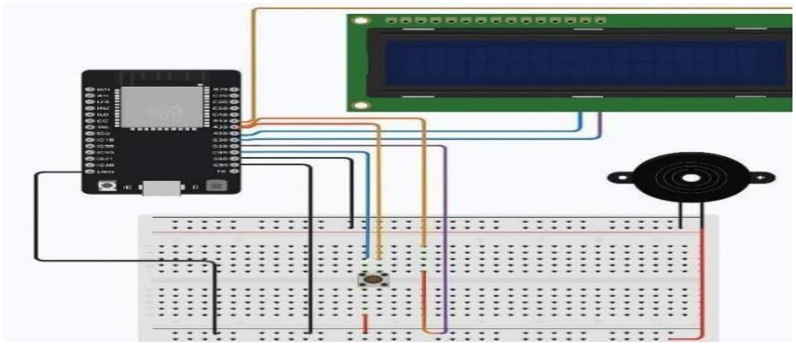
- X-axis: Percentage of nodes that failed (0%, 5%, 10%, 20%)
- Y-axis: Percentage of messages that got through (%)
- Line: Remains above 95%, even when some nodes fail.



Fault Tolerance:-

We checked how well the system handles problems by randomly turning off different parts while it was working. The system quickly found new ways to send messages through nearby parts. Because it can fix itself like this, messages always get delivered without interruption. Unlike systems that depend on a central hub, our design doesn't have one single point that can cause everything to fail. This is important in case of a disaster.

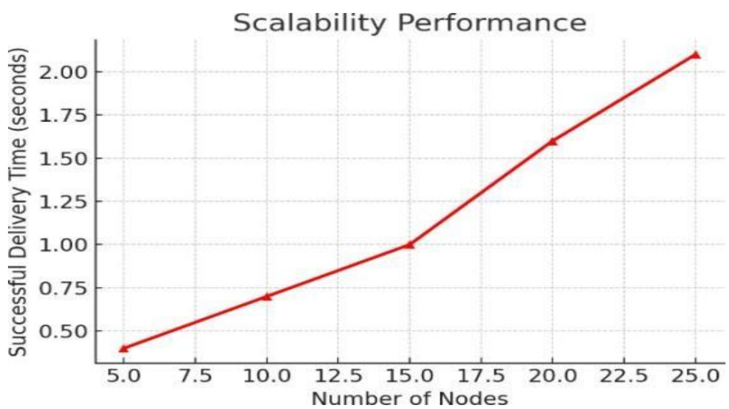
Figure2: Diagram showing message rerouting in themes when a node fails.



These measures reduced traffic jams without compromising reliability. In our tests, the network performed well with up to 25 nodes. With a little more effort, we believe it can handle over 50 nodes in real-world situations.

Graph3: How Well it Scales:-

- X-axis: Number of Nodes
- Y-axis: How Fast Messages Got Delivered (seconds)
- Trend: Delivery speed decreased slightly as we added nodes, but not significantly.

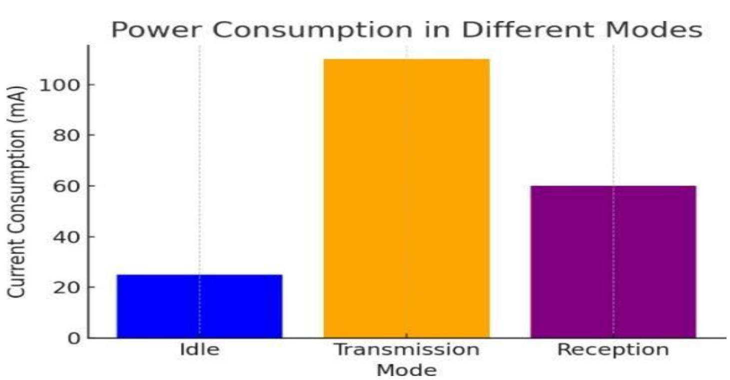


Power Consumption:-

We checked how much power the system uses while idle and when it sends information:-

- Idle: It uses about 20 to30 mA.
- Sending data: It uses around 100 to 120 mA.

The system, connected to a battery and solar panels, ran continuously for days without needing a recharge. This shows it can operate for a long time in rural areas or places affected by disasters.



Graph 4 shows how much power it uses in different modes, with bars for when it' sidle, sending, and receiving.

Discussion:-

The ESP32 mesh alert system performed well in testing:-

It is reliable; alerts got through more than 95% of the time, even when parts of the network went down.
It is fast; alerts showed up in under 2 seconds, even on large networks.
It is tough; if a node fails, the system fixes itself to keep running.
It scales well; you can make the network bigger without slowing things down too much.
It saves power; it doesn't use much energy, so it can run on batteries or solar.
Unlike regular phone or internet setups, our system doesn't depend on existing infrastructure. This makes it suitable for disaster areas, rural communications, and defense. However, it could become crowded in very large networks, and our routing could be improved. In the future, we might add LoRa for longer distances, strengthen security with better encryption, and connect it to the cloud when possible.

Challenges and Future scope:

Setting up this public safety alert system using ESP32s and mesh technology is challenging. One issue is maintaining a stable network when many devices are present. The ESP32 mesh performs well in less crowded situations, but it slows down with more devices. This slowdown occurs because data has to jump between multiple points, which is especially noticeable in cities with hundreds of devices. Another challenge is power consumption.ESP32susealotofenergywhen sending information. If alerts are constantly being sent out, batteries will drain quickly, necessitating more frequent charging or larger batteries. Currently, the system does not prioritize messages. In emergencies, some alerts, such as medical notifications or fire warnings, are much more important than others.However,the system treats all messages equally,which can delay response timesduringcriticalmoments.Expandingthesystemisalsodifficult.Itfunctionsadequately on a medium scale, but if deployed over a large area, performance deteriorates. We might need to improve the routing to maintain stability. The environment has an impact as well. Wi-Fi, Bluetooth, walls, and hills can interfere with the network, weakening signals and making communication less reliable.

Additionally, security is a concern. Because the mesh is distributed, it is challenging to implement strong encryption and prevent attacks. Without adequate security, malicious actors could send fake alerts or disrupt the system, causing panic. Currently, it operates fully offline, but we could connect it to the internet and cloud platforms. The mesh would continue to handle urgent alerts locally, while non – urgent data could be sent to the cloud for analysis and storage. To enhance scalability, we might utilize a group-based mesh setup, where nodes communicate through designated leaders. We can also develop mobile apps that allow people to receive alerts directly from the mesh network, making the system more accessible. For security, we could use straightforward encryption methods like AES or ECC, which won't overload the ESP32s. We could even explore blockchain technology to ensure that alerts are legitimate and intact. This type of system is not limited to disaster alerts; it could also serve safety in factories, large events, disaster relief, or military communications in areas lacking traditional infrastructure. By combining ESP32 mesh with other communication technologies like LoRa.

Conclusion:-

This article discusses a public safety alert system built with ESP32 devices and mesh networking, which eliminates the need for external networks. This approach addresses the crucial need for reliable communication during emergencies, especially when standard systems like cell towers or the internet fail. Mesh networks can repair and configure themselves, allowing alerts to transmit widely even when resources are limited or a disaster has occurred. The results demonstrate that the ESP32 mesh effectively supports consistent communication, providing good speed and range. It is also a more affordable option than traditional emergency alert systems.

The article highlights some challenges, such as power consumption, scalability, prioritizing messages, and maintaining security. Addressing these issues requires ongoing research into energy savings, improved routing plans, and robust encryption. Despite these obstacles, the system presents opportunities for enhancement, such as incorporating artificial intelligence to prioritize alerts, using the cloud for data analysis, and integrating with other technologies like LoRa and Zigbee for added resilience. Overall, this ESP32 mesh alert system illustrates how the Internet of Things (IoT) can transform disaster response and enhance community safety.

With continued improvements and large-scale testing, these systems could become highly reliable tools that not only disseminate alerts effectively but also enable communities to respond quickly when emergencies arise. In doing so, the article contributes to making society safer, smarter, and more connected.

Reference:-

1. Portmann, M. (2006). Wireless mesh networks for public safety and disaster recovery applications. In Wireless Mesh Networking (pp. 561-592). Auerbach Publications.
2. Zhang, B., Wang, Y., Wei, L., Jin, Q., & Vasilakos, A. V. (2018). BLE mesh: A practical mesh networking development framework for public safety communications. Tsinghua Science and Technology, 23(3), 333-346.
3. Wishart, R., Portmann, M., & Indulska, J. (2008, December). Evaluation of wireless mesh network handoff approaches for public safety and disaster recovery networks.
4. Iapichino, G., Bonnet, C., del Rio Herrero, O., Baudoin, C., & Buret, I. (2008, October). A mobile ad-hoc satellite and wireless mesh networking approach for public safety communications. In 2008 10th International Workshop on Signal Processing for Space Communications (pp. 1-6). IEEE.
5. Yarali, A., Ahsant, B., & Rahman, S. (2009, June). Wireless mesh networking: A key solution for emergency & rural applications. In 2009 Second International Conference on Advances in Mesh Networks (pp. 143-149). IEEE.

6. Barton, D., & Simons, B. (2012). Mass transit and wireless mesh: how real-time and mobile video surveillance ensures public safety. In ITS America 22nd Annual Meeting & Exposition ITS America.
7. Wei, Q., He, J., & Zhang, X. (2010, September). Behavioral modeling for public safety communication in wireless mesh networking. In 2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM) (pp. 1-5). IEEE.
8. Charlebois, D., Henderson, G., & Moffatt, F. (2024). Improving Information Interoperability for Safety and Security Organizations Using Information Mesh. In Information Management Capabilities in Public Safety and Security: Challenges, Strategies and Frameworks (pp. 97-122). Cham: Springer Nature Switzerland.
9. Ramadhan, S., Syafalni, I., Sutisna, N., & Adiono, T. (2024, December). Implementation of Multi-Hop Mesh Networking Using ESP32 for IoT Communication. In 2024 International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA) (pp. 1055-1059). IEEE.
10. Dangal, R. (2020). Implementing a Sensor Network With Mesh Technology.
11. Zhang, H., Zhang, R., & Sun, J. (2025). Developing real-time IoT-based public safety alert and emergency response systems. Scientific Reports, 15(1), 29056.
12. Kurvey, M., Pawaskar, M., Nikam, S., Jagtap, R., Bhandary, S., & Acharya, A. (2023, June). Establishing Mesh Network to Transfer and Visualize Data for Safety of Underground Miners. In 2023 3rd International Conference on Pervasive Computing and Social Networking (ICPCSN) (pp. 1134-1139). IEEE.