JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

EDGE COMPUTING SYSTEM FOR REMOTE MONITORING AND CONTROL OF INDUSTRIAL SHOP FLOOR

Anshul Rawtani
Third Year Engineering Student
Department of Electrical And Electronics Engineering
BITS Pilani, Pilani Campus, India

Abstract: This project is an IIoT (Industrial Internet of Things) implementation for remote monitoring and control of industrial shop floors. Edge Computer System in the cloud collects data produced by devices such as sensors. It organizes data into categories and stores them for use by Subscribers. The data is disseminated based on the subscription demands of the Subscribers, who are the control operators. The project framework can be extended for remote monitoring of several shop floors across various industrial units by organizations that provide control and monitoring services.

Key Words: IoT, CIoT, IIoT, Edge Computing System, MQTT, MQTT Broker.

1. INTRODUCTION

The Internet of Things (IoT) refers to the network of physical items embedded with sensors, software, and other technologies to connect and exchange data with other devices and systems over the internet. These gadgets range from simple domestic items to complex industrial equipment. Today, there are more than 10 billion Internet of Things (IoT) devices that are connected; by 2025, this number is expected to reach 22 billion. (Oracle Portugal. What is internet of things.)

Over the years, IoT has evolved and today there are many types of IoT based on its use. (Duggal, N. 2022)

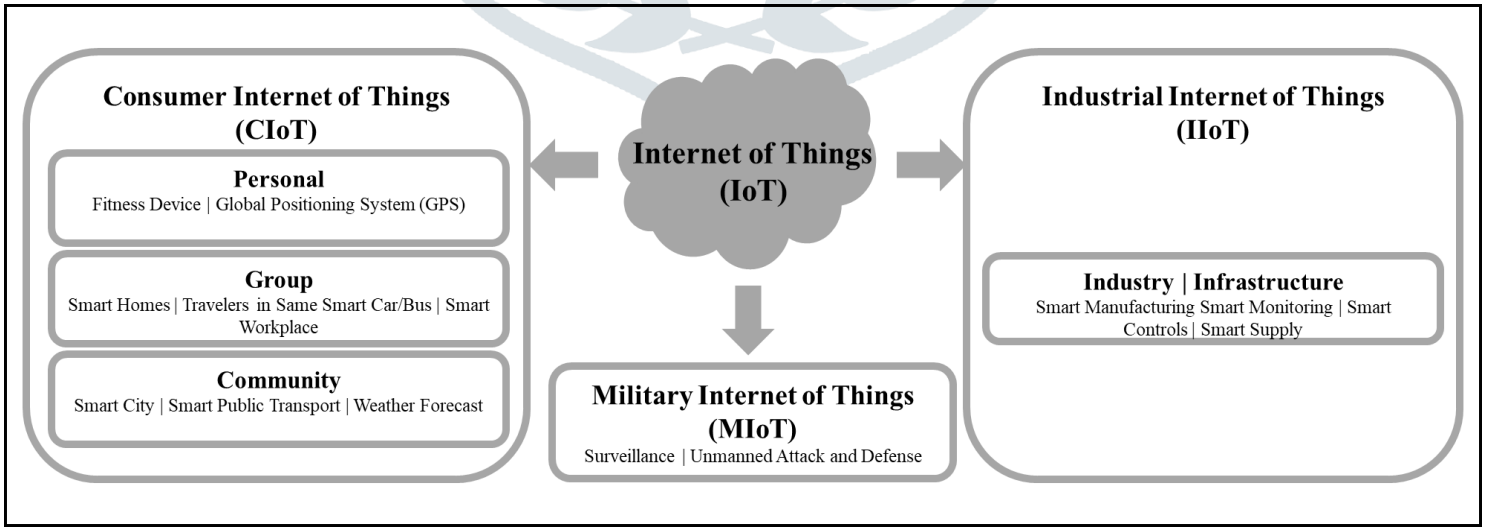

Figure 1: Types of IoT.

Consumer Internet of Things (CIoT) is widely used today. The use has been accelerated in recent years due to access to low-cost, low-power sensor technology, better connectivity, Cloud Computing Platforms, and Machine Learning and Analytics.

When it comes to the Industrial and Infrastructural Ecosystem a more production, supply, and availability-oriented system is required. For this Industrial Internet of Things (IIoT) is considered. IIoT can be referred to as the system of interconnected devices such as sensors, controllers, and actuators which enables the collection of data from these devices from all locations. Data flows into the cloud

data centers where data analysis and analytics are done to make the overall ecosystem smarter. Some common Use Cases for IIoT are (Taylor, K. 2020):

- Smart manufacturing.
- Smart power generation and distribution.
- Smart gas or water sourcing distribution.
- Smart connectivity and bandwidth.
- Smart digital supply chains.
- Predictive maintenance to reduce leakage, downtime, and outage.
- Air and train travel safely.
- Auto recovery mechanism for higher availability.
- Connected logistics.

The main aim of the project was to build an edge computer using Python on Linux. This edge computer acts as a gateway between the sensor node network and the cloud servers and helps to store, transmit and process data effectively. The application layer IoT protocol that has been used to implement edge computing is the MQTT protocol. The edge computer will be used in an IIoT context on an industrial shop floor.

2. LITERATURE REVIEW

Edge Computing:

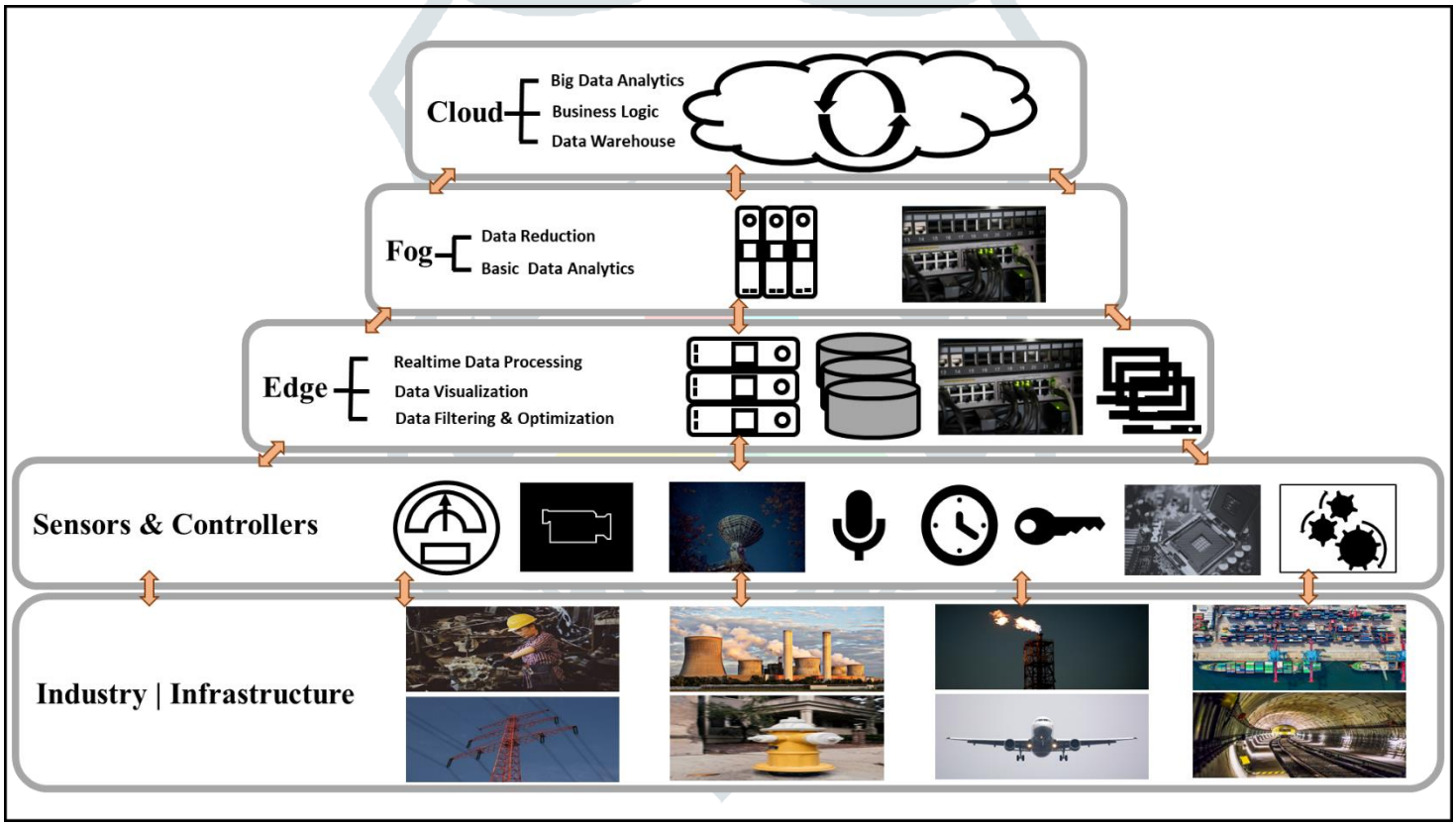

Figure 2: Typical IIoT Edge Computing Architecture.

Edge computing is a distributed computing system that connects enterprise applications to data sources such as IoT devices via local edge servers. It is a network strategy that strives to bring processing power, memory, and storage as close to end users as possible. (IBM. What is Edge Computing) It is a decentralized and distributed computing in which the information processing is located closer to the edge. This proximity to data at its source can deliver strong business benefits: faster insights, improved response times, and better bandwidth availability. (Mohanan, R. 2022) The edge refers to the locations where the network's server can deliver the computing functionalities to the customers most expediently. The idea of moving computing services closer to service users or data sources is at the core of edge computing. (HEAVY.AI. What is Edge Network) Edge computing helps address the major concerns of bandwidth, latency, reliability, and data sovereignty. (Bigelow, S. J. 2021) It is driven by growing use cases like IoT, AR/VR, robots, machine learning, and telecom network activities that demand service provisioning closer to customers. (Vermesan, O. et al.2022) It completes the hybrid computing architecture, which allows for the utilization of centralized computing for computationally demanding workloads and edge computing for workloads that call for near real-time processing.

Putting all computing tasks in the cloud has turned out to be a good way to process data since the cloud has more computing power than the things at the edge. With the amount of data being made at the edge increasing, the speed of moving data is becoming the bottleneck for cloud-based computing. (Shi, W. et al.2016) Systems having real-time usage like self-driven systems cannot afford delays in data transmission. In an autonomous vehicle, for example, one Gigabyte of data will be generated by the car every second

and it requires real-time processing for the vehicle to make correct decisions. (Cheruvu, Ria. 2015). If all the data needs to be sent to the cloud for processing, the response time would be too long. Not to mention that current network bandwidth and reliability would be challenged for its capability of supporting a large number of vehicles in one area. In this case, the data needs to be processed at the edge for shorter response time, more efficient processing, and smaller network pressure. (Shi, W. et al. 2016)

Almost all kinds of electrical devices will become part of IoT, and they will play the role of data producers as well as consumers, such as air quality sensors, LED bars, streetlights, and even an Internet-connected microwave oven. (Pramod K., Aparna C G., 2022) It is safe to infer that the number of things at the edge of the network will develop to more than billions in a few years. Thus, the raw data produced by them will be enormous, making conventional cloud computing not efficient enough to handle all this data. This means most of the data produced by IoT will never be transmitted to the cloud, instead, it will be consumed at the edge of the network. (Shi, W. et al. 2016) IDC predicts that by 2023 more than 50% of new IT enterprise infrastructure deployed will be at the Edge rather than in Corporate Data Centres up from less than 10% in 2020. (IDC. IDC Table of Contents. (n.d.))

2.1. Practical Use Cases of Edge Computing:

- Sensors in valves at a petroleum refinery that detect a dangerously high level of pressure in the pipes.
- Sensors that monitor the soil moisture content and accordingly take action.
- Autonomous cars that detect blockages.
- Health care devices that monitor vital signs and respond accordingly.
- A camera that performs local vision recognition.

In all these use cases the key is “latency” and “speed of response”.

2.2. Key drivers of edge computing:

- Huge data being generated.
- 5G+IOT.
- Faster processing.
- Intelligence in devices.
- Low latency.
- Concerns related to data sovereignty, security, and privacy.

2.3. Key Components of Typical Edge Computing System:

Edge Client: It captures all types of data produced by various client devices. Every IoT device with some level of built-in intelligence and storage has it installed, and it communicates with the nearby Gateway Server.

Edge Gateway Servers: These are situated physically very close to the Edge Client devices in order to decrease latency and boost processing power for heavier workloads.

Edge data Centres: End users receive cloud computing resources and cached material from these more compact facilities, which are situated near the communities they serve. Usually, they are connected to a sizable central data centre or a number of data centres.

2.4. MQTT Protocol

Message Queue Telemetry Transport: MQTT is an easy-to-use and lightweight communications protocol for connecting a wide variety of gadgets. It's a publish-subscribe protocol that runs on top of TCP. This protocol is well suited for data transmission between low-power, low-bandwidth devices. The IoT Framework relies heavily on this message protocol for its communication needs.

Publish-Subscribe Model: For this architecture, there is no need to set up a direct link between the many clients who communicate with one another. All client-to-client interactions take place solely through a middleman, or "Broker," in all cases.

MQTT Client and Broker: To the broker, users send their communications on a variety of subjects. The broker acts as a centralized server that takes in these communications and sorts them out according to their respective themes. This information is subsequently relayed to the clients that have signed up for that particular issue. So, if a client is subscribed to a certain topic, they will get any messages sent to that subject. (GeeksforGeeks. 2021)

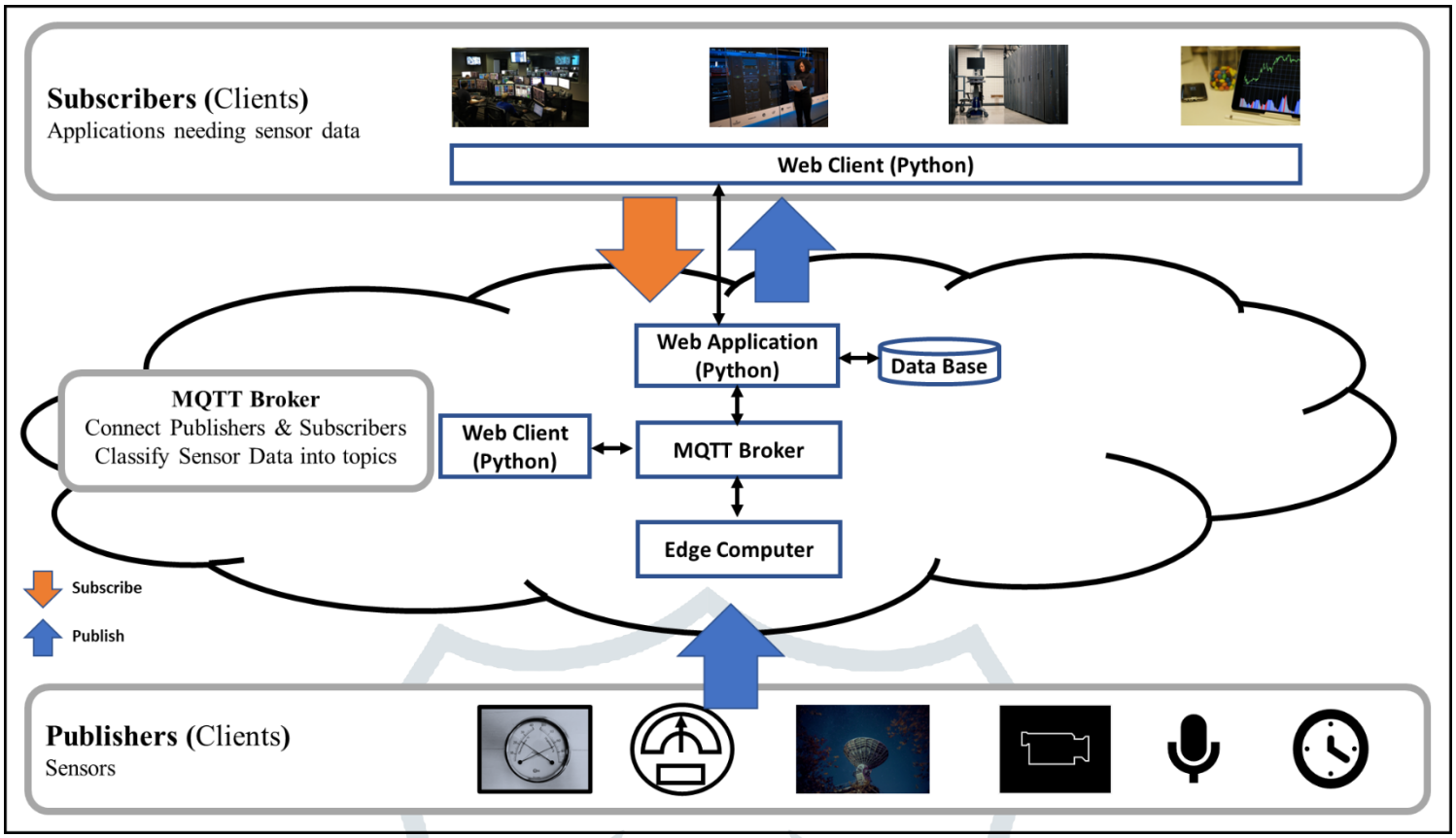

Figure 3: MQTT Architecture.

3. PROJECT OVERVIEW AND IMPLEMENTATION DETAILS

3.1. Project objectives

The project is to design a system which helps in remotely monitoring and controlling the overall working condition of a typical industrial shop floor. Typical industrial shop floor can have furnaces or boilers, rotating machines, gas cylinders etc. The parameters like temperature, rmps, pressure, humidity, etc. have to be monitored on a real-time basis so that appropriate steps to control these parameters as per need, can be taken. These parameters are measured by various types of sensors. The sensors are connected to a network and the data generated by these sensors can be transmitted via the network to servers for collection, processing, and storage. The control operators who work remotely can request for the type of parameter they wish to monitor, and what frequency they want the parameter to be obtained and also send control instructions to change the settings of various equipment so that the parameters are at the desired levels.

The control operators may not be on the dedicated network of the industrial unit whose shop floor is being monitored. Control operations may also have been outsourced to a third party that operates remotely. The systems may also not only be catering to the requirement of one shop floor of any one industrial unit. The system may be catering to multiple shop floors of multiple industrial units with control operators accessing through the internet.

For this project, I am considering one industrial shop floor. The core system is hosted on a cloud. The sensors collect and send data to the core system over the internet and not through a dedicated private Wide Area Network (WAN). The control operators also access the core system over the internet and not through dedicated private WAN.

The sensors that capture the parameters like temperature, rmps, pressure, humidity, etc. are the “Publisher Clients”. The systems and monitoring terminals used by the control operators are the “Subscriber Clients”. The system which ensures communication between the “Publisher Clients” and “Subscriber Clients” without the two types of clients having any direct connection is the "Broker”.

The above requirement can be met by an Edge Computing System that has the following:

- Mechanism to transmit data collected by the sensor nodes.
- Mechanism to collect, store and process the data.
- Mechanism to relay commands to the sensor nodes.
- Facility for the control operator to request for the subscription of the type of data along with the frequency.
- Facility to enter the commands that would then be finally relayed to the sensor nodes.

3.2. Data:

The system needs to collect, store and process the following type of data:

3.2.1. Master Data:

- **Clients:** Subscriber Clients (control operators/applications used by control operators) and Publisher Clients. (sensor nodes)
- **Topics:** Data pertaining to different types of equipment like furnaces, rotating machines, gas cylinders, etc.
- **Type of data pertaining to each topic:** E.g., Temperature, Pressure, RPM, etc.
- **Type of control commands available for each topic.:** stop, increase, decrease, etc.

3.2.2. Transactional Data:

- **Generated Data:** Actual Data generated for topic and type, by sensor nodes.
- **Subscription details:** Type of data that each subscriber has subscribed to and the frequency at which each subscriber needs the data. For E.g., Control Operator has asked to send the collected temperature data once every 10 minutes.
- **Control Command Details:** Commands submitted by each subscriber e.g., Control Operator wants the Temperature Sensor Node to capture temperate data every 1 minute.

3.3. Menu Options:

The control operator has the below-mentioned Menu Options available:

- **View list of subscribed topics**: The menu allows the Control Operator who is the Subscriber Client, to view the list of current topics which the system supports. This list will get dynamically updated based on additions/removals of sensor nodes, topics, and the data pertaining to each topic.
- **Subscribe to a topic:** The menu allows the Control Operator which is the Subscriber Client to select and subscribe to a Topic and the type of data related to the topic. It also allows the Control Operator to issue control commands.
- **Un-subscribe a topic:** The menu option allows the Control Operator to un-subscribe a topic.
- **Defining a schedule:** Control Operator is prompted to set up a schedule. The Edge Computing System will receive data from the sensors as per this schedule.
- **Publish sensor data**: The data of all the sensors is received and then stored in the sensor database. Using this feature the sensor data from the sensor database is published on a particular topic.
- **Publish sensor data as per user-defined schedule**: The Edge Computing System publishes the data to the required Subscriber Client connected to the Edge Computing System in JSON format. The published data is also stored in the sensor database for future reference.
- **View queued messages from a subscribed topic:** This feature is useful to view the queued messages that were received from the subscribed topics when the system was offline. All messages are preserved and as soon as the system is back online the Subscriber Client will be able to view all these messages that were queued.
- **Exit:** To terminate a program the Subscriber Client can choose to exit at any point

3.4. Software Prerequisites and Databases:

- **Operating System:** Linux
- **Programming Language:** Python
- **IDE:** Visual Studio Code
- **Other additional Software:** MQTTBox, Virtual Serial Port Driver, Realterm serial terminal
- **Databases:**
  - **Masters:**
    - **clients.csv:** It stores the Subscriber Clients (control operators/applications used by control operators) and Publisher Clients (sensor nodes)
    - **commands.csv.** It stores the Type of control commands available for each topic.
    - **OptionsforTopicSubs.csv:** It stores the Topics: Data pertaining to different types of equipment like furnaces, rotating machines, gas cylinders, etc. It also stores the type of data pertaining to each topic: E.g., Temperature, Pressure, RPM, etc
  - **Transactional:**
    - **sensordatabase.csv:** Actual Data generated for topic and type by sensor nodes
    - **DataTopics.csv:** It stores Subscription details i.e. the Type of data that each subscriber has subscribed to and the frequency at which each subscriber needs the data. For E.g., Control Operator has asked to send the collected temperature data once every 10 minutes.
    - **ControlCommandsdb.csv:** It stores the Control Command Details: Commands submitted by each subscriber e.g., Control Operator wants the Temperature Sensor Node to capture temperate data every 1 minute.

3.5. Program components

The program components make use of Python libraries for mqqt client, pandas, csv, schedule, threading, json. The components are capable of publishing a single message, scheduling messages, subscribing to a topic, unsubscribing, and displaying saved sensor data. The components make use of the data that comes from sensors in json string format and convert it to pandas data frame and add it to the sensor database in csv format.

3.5.1. Main Program

Below are the important functions:

Function selecting_options()

- Presents the User with a choice amongst the multiple topics of Pressure, Temperature, rpm, Humidity etc to either publish or subscribe a message.

Function schedule_function(Topic)

- Presents the User with 2 options of choosing to schedule a message either minute-wise or second-wise.
- Subsequently asks for the number of minutes or seconds for scheduling.
- Makes use of the scheduling library functions.
- Invokes function of the respective topic for publishing the message.
- Any one of the functions pubschedTopic1(), pubschedTopic2(), pubschedTopic3() etc are invoked.
- These functions publish sample messages to the respective topic.
- The messages are published as per the schedule.

Function on_message(client, userdata, msg)

- This function is invoked from an event listener whenever a message arrives onto a topic
- The message request is read from the topic
- This function decodes messages from the sensors and adds it to the database
- Along with the topic name and the current timestamp, the message is appended to a csv file

Function run_continuously(interval=1):

- This uses a thread that runs continuously but checks at the specified interval for any pending jobs that are to be executed.
- The thread will sleep for a specified interval.

Function publish_msg(topic)

- This function asks the User for a message.
- It sends the message to the specified Topic.

Function database_view()

- It presents the user with the option of the topic for which the data needs to be displayed.
- It then reads the data from the csv where all sensor data is stored.
- The filtered data frame is accordingly populated from the specified topic and the data is presented.

Function viewoffline_msg()

- It converts csv file data to the list format.
- Reconnects to the topic which has already been subscribed and also prints the queued messages.

Function unsubscribe_topic()

- Converts the csv database to the list format
- It checks whether there are any subscribed topics available or not.
- If the topic is subscribed it prints the topics to which the user has currently subscribed to.
- User enters the choice of the topic to which he wants to unsubscribe. The topic is then removed from the list and the csv file is cleared completely.

Function subscribe_topic()

- It converts csv database to list format and checks if there are any subscribed topics available or not.
- Based on the user choice of topic, the required topic is unsubscribed.

Function publish_msg(topic)

- It publishes messages to the client and displays the sensor data in json format.
- It converts the csv file data to json format in records orientation and publishes messages to the required topics.
- It also shows the confirmation message that the message has been published.

4. FUTURE SCOPE

The advancements in Machine Learning and Artificial Intelligence will expand the application of IIoT in various sectors. Self-driving cars are currently undergoing trials and could soon become the future of the automotive industry. Delivery and passenger drones are under trial and their operational control is a very big case for IIoT. Agriculture in a country like India can be revolutionized by the adoption of IIoT. Precision farming where crop fields are observed by monitoring through sensors for soil moisture content, humidity, temperature, etc along with agricultural drones deployed could increase agricultural productivity. Another area of application is the healthcare industry where smart wearables used by patients can be used to track and monitor their health conditions even remotely. India today can boast of IIoT, Robotics, and automation coming together to build the so-called Food-as-a-service (FAAS) and churn out machine-made biryanis from multiple cloud kitchens whose inventory and equipment can be remotely monitored. The future looks very promising and full of possibilities.

5. CONCLUSION

This project is a basic application of IIoT in which sensors are deployed in a factory setting, the sensors transmit data and this data is then processed to monitor and track the operations. The further potential of this would be to utilize the data for predictive maintenance and also to develop mechanisms to detect faults, trigger alarms and increase factory automation. This program can be extended for use by service-providing Brokers that provide the Edge Computing System as an industrial monitoring and control service. Multiple industries with multiple shop floors can be the Publishers. Companies providing industrial monitoring and control services can be the Subscribers.

6. REFERENCES

1. Bigelow, S. J. (2021, December 8). *What is edge computing? Everything you need to know*. Data Center. https://www.techtarget.com/searchdatacenter/definition/edge-computing
2. Cheruvu, Ria. (2015). Big Data Applications in Self-Driving Cars. 10.13140/RG.2.2.12302.05445.

3. Duggal, N. (2022, November 22). *What Are IoT Devices : Definition, Types, and 5 Most Popular Ones for 2023*. Simplilearn.com. https://www.simplilearn.com/iot-devices-article
4. GeeksforGeeks. (2021, December 24). *Introduction of Message Queue Telemetry Transport Protocol (MQTT)*. https://www.geeksforgeeks.org/introduction-of-message-queue-telemetry-transport-protocol-mqtt/
5. *What is Edge Network? Definition and FAQs | HEAVY.AI*. (n.d.). https://www.heavy.ai/technical-glossary/edge-network. Accessed on December 26,2022.
6. *What Is Edge Computing*. (n.d.). IBM. https://www.ibm.com/cloud/what-is-edge-computing. Accessed on December 26,2022
7. *IDC.com IDC Table of Contents*. (n.d.). IDC: The Premier Global Market Intelligence Company. https://www.idc.com/research/viewtoc.jsp?containerId=US44747919
8. Mohanan, R. (2022, February 10). *What Is Edge Computing? Components, Examples, and Best Practices*. Spiceworks. https://www.spiceworks.com/tech/edge-computing/articles/what-is-edge-computing/
9. "What Is the Internet of Things (IoT)?" *What Is the Internet of Things (IoT)? | Oracle Portugal*, www.oracle.com/pt/internet-of-things/what-is-iot. Accessed 25 Dec. 2022.
10. Pramod K., Aparna C G., 2022. EDGE COMPUTING LOT FROM HPE INTELIGENCE EDGE. *International Journal of Creative Research Thoughts. https://ijcrt.org/papers/IJCRT2205564.pdf*

11. Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge Computing: Vision and Challenges. *IEEE Internet of Things Journal*, *3*(5), 637–646. https://doi.org/10.1109/jiot.2016.2579198
12. Taylor, K. (2020, July 8). *8 Applications of Industrial Internet of Things*. HitechNectar. https://www.hitechnectar.com/blogs/industrial-internet-of-things-applications/
13. Vermesan, O., Eisenhauer, M., Serrano, M., Guillemin, P., Sundmaeker, H., Tragos, E. Z., Valiño, J., Copigneaux, B., Presser, M., Aagaard, A., Bahr, R., & Darmois, E. C. (2022). The Next Generation Internet of Things – Hyperconnectivity and Embedded Intelligence at the Edge. *Next Generation Internet of Things – Distributed Intelligence at the Edge and Human-Machine Interactions*, 19–102. https://doi.org/10.1201/9781003338963-3