Software Engineering Department

Braude College

Capstone Project Phase A

An Integrated Internet of Things Monitoring System – Joint work with TAMK University

By:

Dmitry Chudnovsky

Or Benyamin

Advisor:

Dr. Naomi Unkelos Shpigel

Project Code:

24-2-D-11

Link to GitHub:

https://github.com/ChDimitry/FinalProjectBraude

Table of Contents

Abstract

In recent years, the Industrial Internet of Things (IIoT) has emerged as a transformative force in modern industrial processes, revolutionizing data collection, analysis, and utilization across manufacturing and research environments. This project addresses a specific challenge faced by TAMK, which operates a laboratory equipped with autonomous robotics featuring various types of sensors. The main challenge of the laboratory lies in the absence of a comprehensive system to effectively monitor and transmit live data from these diverse robotic units.

To address this challenge, we develop a web application that connects to servers and interfaces with the robotic systems. This solution creates a unified platform for data collection, monitoring, and analysis. The system shall display and monitor real-time data collected from the sensors on the robotics, providing users with the ability to visualize and analyze sensor data, track changes over time, and gain insights into the system's performance and conditions.

The primary contribution of this system lies in its capacity for remote communication and display, serving as a proof of concept for an inter-institutional IIoT framework. This project emphasizes cooperation between TAMK and Braude, highlighting the potential for collaborative efforts in the realm of Industry 4.0. Throughout the development process, we maintained regular communication with the project manager, incorporating his feedback to ensure the system met all requirements and expectations.

By employing Agile methodologies, we would be able to adapt to changing needs and deliver a robust solution. Notably, this project marks a significant milestone as the first Industry 4.0 initiative undertaken at Braude, paving the way for future advancements in industrial automation and data-driven decision-making. The collaboration between the two institutions demonstrates the potential of inter-institutional partnerships in driving innovation in IIoT and setting a foundation for future developments in the field.

1. Introduction

Tampere University of Applied Sciences (TAMK) in Finland stands at the forefront of innovation in applied sciences and technology. At the heart of its cutting-edge facilities lies a state-of-the-art robotics laboratory, serving as a microcosm of the future smart factory. This advanced facility houses an impressive array of autonomous robotic units, each equipped with various sensors capable of collecting real-time data on different aspects of their operation and environment. The laboratory provides an ideal setting for students, researchers, and industry partners to explore the integration of robotics, sensors, and data analytics in conditions that closely mimic actual manufacturing environments. Despite its advanced equipment, TAMK's robotics laboratory faced a significant challenge: the absence of a comprehensive system to effectively monitor and transmit live data from its diverse robotic units. This limitation hindered the full realization of the laboratory's potential as a testing ground for Industrial Internet of Things (IIoT) applications. To address this, TAMK initiated a collaborative project with Braude College of Engineering in Israel, aiming to develop an integrated IIoT software system. This book documents the journey of developing this system, exploring the challenges faced, the solutions devised, and the lessons learned throughout the process. By bridging the gap between academic research and practical application, this project not only enhances the capabilities of TAMK's robotics laboratory but also serves as a proof of concept for inter-institutional IIoT frameworks in the realm of smart manufacturing and research facilities.

2. Literature Review

2.1 Digital Twin

Digital Twin technology represents a transformative advancement with potential applications spanning manufacturing, healthcare, and smart cities. There are evidences that emphasize its promise in enhancing performance and enabling predictive maintenance.

On one hand, we can understand that data integration and security are crucial for realizing the full benefits of Digital Twins [2]. On the other hand, there is a necessity for a standardized definition to alleviate confusion and promote consistent understanding across industries [10].

Together, these insights stress the importance of overcoming technological and definitional hurdles to fully leverage the capabilities of Digital Twin technology.

2.2 Sensors and IoT

The integration of IoT and advanced sensor technologies is revolutionizing robotics in agriculture and industry.

Autonomous farming robots, equipped with various sensors and image processing capabilities, use IoT and wireless sensor networks to detect weeds, monitor environmental parameters, and navigate fields, enhancing precision and efficiency while reducing costs [5]. Effective decision-making in IoT applications relies on data processing methods like denoising, outlier detection, and the integration of cloud, fog, and edge computing [6].

In industrial robotics, sensors such as tactile, visual, laser, and inertial are vital for tasks like human-robot collaboration and navigation [7]. Advancements in these sensors and data processing algorithms are making robots more intelligent and independent, despite the challenges of meeting increasing performance demands.

The integration of IoT and advanced sensors in agriculture and industry is enhancing robotic precision, efficiency, and intelligence while reducing costs, though challenges in performance optimization remain.

2.3 IoT and Robotics (IoRT)

The integration of the Internet of Things (IoT) with robotics represents a rapidly evolving area with significant technological and practical implications.

There are two which explore the integration of IoT with robotics, highlighting its technological implications and real-world applications. The first, delves into the feasibility and challenges of IoT-aided robotics, noting issues with protocols, communication technology, and security, and emphasizing the need for further research despite the technology's maturity [3] . The second, provides a comprehensive overview of existing concepts and architectures, underscoring the benefits and challenges of IoT in robotic systems [4].

Additionally, integrating robotics with IoT was explored to form the Internet of Robotic Things (IoRT) [9]. It provides an understanding of IoRT's principles, proposes new frameworks, and highlights key research challenges. In a conducted research, the authors conclude that while IoRT holds significant potential, it presents unique technical challenges that need to be addressed [11]. By combining sensing, AI, and IoT, IoRT aims to create smart, autonomous systems, significantly advancing IoT by integrating AI, robotics, and machine learning [ibid]. These insights illustrate that while IoT and IoRT-enhanced robotics offer significant potential, addressing technological and research gaps is essential for advancing the field.

Together, these insights illustrate that while IoT-enhanced robotics offers significant potential, addressing technological and research gaps is essential for advancing the field.

2.4 Real Time Monitoring in IoT

Real time monitoring and control systems are vital in environmental and industrial applications due to their ability to provide immediate data and feedback, enabling quick decision making and response.

One example of a robot for real time environmental and industrial monitoring is the “InterBot 1.0” [8]. The purpose of such a robot is to leverage advanced sensor technology for measuring temperature, humidity and gas levels. It features real time data visualization and allows for continuous monitoring and analysis.

The ability of providing immediate data and feedback enables rapid responses to emerging issues, supports predictive maintenance, and enhances overall system efficiency and reliability.

2.5 IIoT

The Industrial Internet of Things (IIoT) refers to the network of interconnected sensors, instruments, and devices used in industrial applications such as manufacturing, energy management, and logistics. These devices collect and analyze data in real time to enhance operational efficiency, productivity, and decision-making.

There are some advantages for IIoT which include enhanced efficiency and productivity, improved data collection and analysis, cost saving, remote monitoring and control [1]. However, the implementation of IIoT is not without challenges, particularly concerning security and privacy. IIoT systems are susceptible to cyberattacks and hacking, compatibility issues, lack of standardization, and high implementation costs, all of which pose significant barriers to widespread adoption [ibid]. Despite these disadvantages, the potential of IIoT to drive innovation and maintain a competitive edge in the industrial sector is immense.

In conclusion, while the Industrial Internet of Things (IIoT) provides significant benefits like enhanced efficiency, productivity, and cost savings, it also faces challenges in security and privacy. Addressing these issues through robust privacy measures and standardized protocols is essential for industries to fully realize IIoT's potential and maintain a competitive edge in the technological landscape.

3. Research

3.1 Scheduled Meeting

During our work in the semester we had scheduled meetings with Senior Lecturer Mr. Kari Naakka on behalf of TAMK University. Holding meetings with TAMK University on a regular basis constituted an important and main focus during our work. During these meetings we introduced the work that was done and the progress we had in that period of time. Moreover, these meetings contributed to our understanding of topics and points that were not clear to us. At the end of every meeting we agreed on a date for the following meeting and the progress we are going to have for that meeting.

3.1.1 Meetings Summary

3.1.1.1 Meeting Number 1 - 17.6.24

In this meeting we held the first update on the process being done until that point. We introduced a first look on the website, a basic website without map included. We talked about the device's positions on the site and how to transfer this using the API. Furthermore, we solved the issue we had regarding connecting to TAMK’s server using the OpenVPN application. We asked TAMK’s representative about a single API which gathers together all the devices and agreed to get an update until the next meeting.

3.1.1.2 Meeting Number 2 - 27.6.24

In this meeting we held an update on what has been done so far. We have shown a preliminary prototype of the site as for now, which includes 3 devices on a map and when you click on a device, a menu opens where you can see the name of the devices and its features. When clicking on an attribute, you can place it at the top of the list, copy its content and create a visualled graph for it. During the meeting few points were suggested by the customer to think ahead:

- Scaling: choose a time frame.
- Real time meter: bar or needle for the data being displayed.
- Showing Automatic update of data by the website.
- Settings file: file which contains the beginning location, floor plan for more dynamic interface.

The customer mentioned a point to look ahead when the website is complete, having a 3D model.
We ended the meeting by scheduling the next meeting to the 5th of August where we will show the progress in the Phase A Book project.

3.1.1.3 Meeting Number 3 - 5.8.24

In this meeting we first show the customer the progress we had on the book project. We showed him the sections we have done until the meeting and asked him for feedback. The customer was very excited to see the progress we have done. As a result, he asked for few things:

- The customer asked us to send him a copy of the book we have done so far. We agreed to send him a copy of the final book once we finish working on that.
- The customer told us that the name of the system is up to us and we can decide how to call the system.
- The customer asked us to deploy the current website and shared with him the link for him to show the progress to other members in the TAMK university and could have a feeling on the website.

At the end of the meeting we agreed on a date for the following meeting.

3.1.1.4 Meeting Number 4 - 11.9.24

In this meeting we first talked with the customer about the comments he had on the copy of the book. The customer emphasized the importance of mentioning him with the words “The customer” and not his private name except for one time when introducing him. In addition, he suggested thinking about different names for the project that will replace the words “software system” with something more specific. The customer also suggested changing the way the requirements are presented from “high level requirements” to “low level requirements”.
Then we started talking about the website itself and the changes we had from the last meeting. We explained that changing the structure of the website required us to start from the beginning and therefore there are some changes from the last meeting. We asked the customer about the 3D model that required to be shown on the website and he informed us that the system responsible for taking pictures of the devices is still not available and will be ready only at the end of the year (see picture 1 in section 10.1).
We agreed on few points to be updated:

- The customer will update us in the next 2 weeks when the data of the robot’s position will be ready.
- Considering the situations where a robot moves fast and slow, and the way it will represent on the map in the website.
- Thinking of a way of comparing 2 sensors' data in the same page. At this point the page will display only the data and will create a graph of a specific device. The customer suggested improving this by enabling it for different devices at the same time and option for different scales for each graph..
- The customer will update us when the data on the 3D model will be available.

At the end of the meeting we agreed on November 6th as our next meeting.

4. Engineering Process

4.1 Development Process

In this section, we will describe our workflow, the methodologies and technologies used in order to build a system which is capable of monitoring and analyzing data collected from various types of robots. This includes data processing, backend frameworks for handling data streams, and frontend technologies for creating dynamic, real-time visualizations. Our approach balanced the need for robust, scalable data handling with flexible, intuitive user interface, ensuring a final product that could adapt to different robot types while remaining powerful and user-friendly.

4.2 User-Centered Design Approach

Our project employs a user-centered design (UCD) approach, placing the customer's needs and requirements at the forefront of our development process. This methodology ensures that the final product not only meets technical specifications but also aligns closely with user expectations and usability requirements.

4.2.1 Regular Customer Meetings

As detailed in our meeting summaries (sections 3.1.1.1 to 3.1.1.4), we conduct regular meetings with the customer. These meetings serve as crucial touchpoints for:

a) Presenting progress and gathering immediate feedback

b) Clarifying requirements and expectations

c) Discussing challenges and potential solutions

d) Planning next steps and setting priorities

4.2.2 Prototype-Driven

We developed and presented prototypes to the customer at various stages:

a) Initial prototype presentation (Meeting 1 - 17.6.24)

b) Updated prototype with customer-requested features (Meeting 2 - 27.6.24)

4.2.3 Conclusion

By adopting a user-centered design approach, we ensure that our solution not only meets technical specifications but also provides a seamless, intuitive experience for the customer. This approach has been instrumental in shaping our development process and will continue to guide our efforts as we move forward with the project.

4.3 Workflow

As shown in figure 1, the project began with an initial meeting with our advisor, where we focused on understanding the project requirements. This step was crucial in defining the scope, objectives, and technological needs of the project. Once the project requirements were clear, we conducted a comprehensive literature review using resources like Google Scholar to gather relevant research and methodologies to inform the project direction.

With the necessary research completed, we decided on the project's architecture, defining the system design and choosing the technology stack to implement. This led us to the next phase, where we started working on the prototype.

Following the prototype's initial development, we held the first meeting with the customer to present our progress and gather feedback. Based on this feedback, we implemented the customer's comments and made necessary revisions to the prototype.

We then organized a second meeting with the customer to showcase the updated prototype. With the customer's approval of the prototype, we moved forward to create detailed UML diagrams using Visual Paradigm. These diagrams, including use case and activity diagrams, illustrated the system's user interactions and workflow processes.

After creating the UML diagrams, we met with the customer for a third time to present the progress of both the book and the system. During this meeting, we discussed project testing strategies to ensure the system's functionality and performance.

Once the book project was completed, we closed that phase and held a fourth meeting with the customer. During this meeting, we presented the final book and demonstrated the website's progress. Furthermore, once the website was fully operational, we conducted group research to assess its functionality and usability. After addressing all comments, we published both the website and the book project.

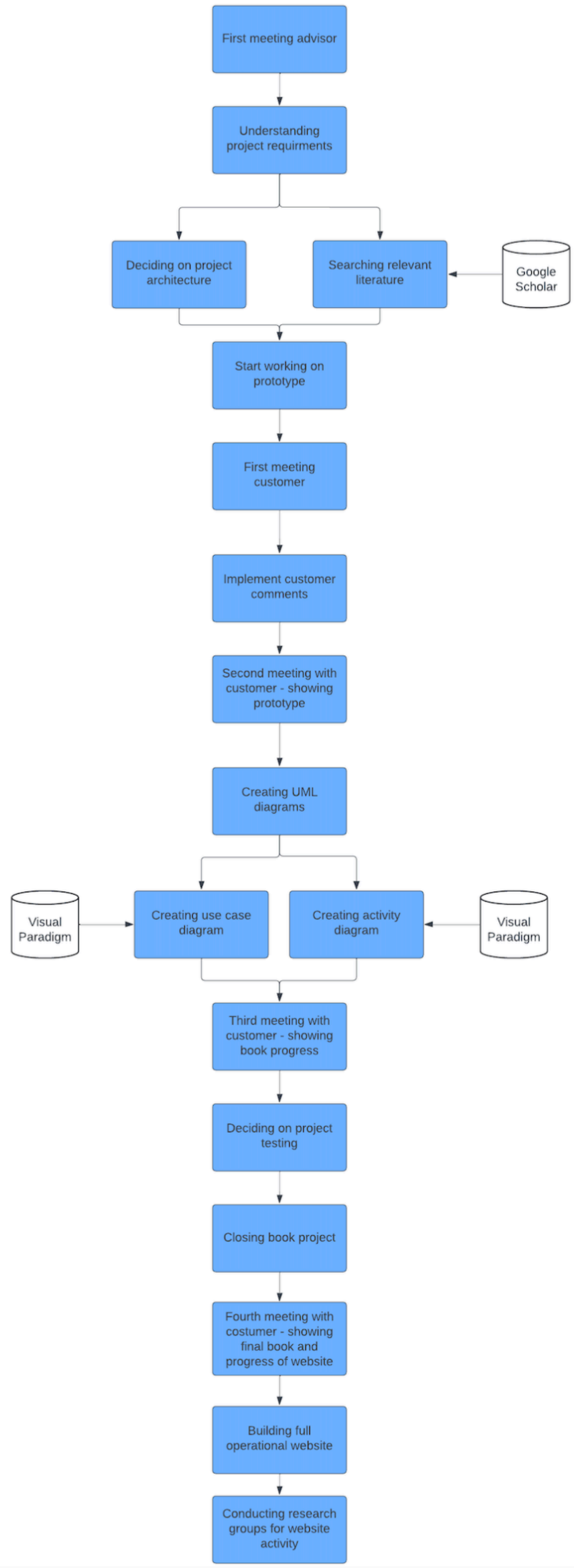


figure 1. Workflow diagram

4.3 Architecture Diagram

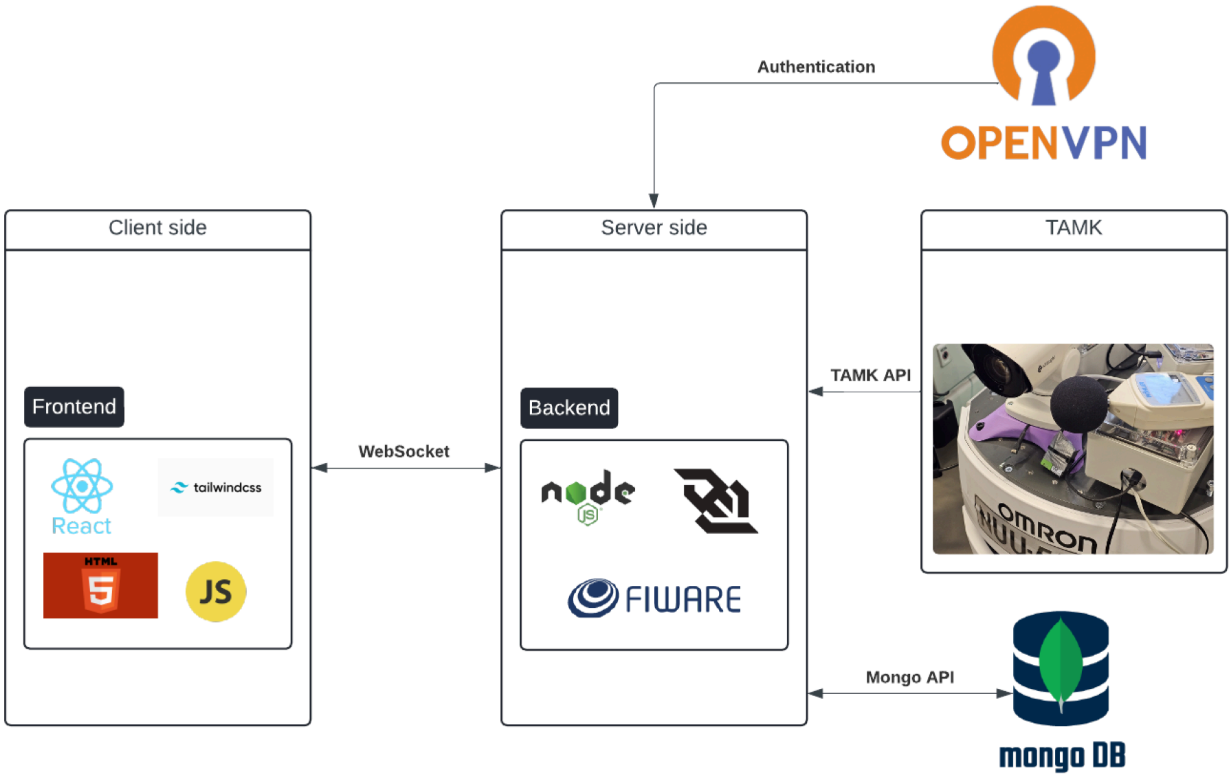


figure 2. Architecture diagram

The diagram (figure 2) illustrates our architecture which is combined by two main components: client side and server side. The client side is built on 4 main parts which are: react, tailwind, html and javascript. The server side is built on 2 main parts which are the nodejs and Fiware. We are using mongoDB for user management and OpenVPN for accessing API.

4.4 Technologies Review

4.4.1 Websocket

WebSocket is a bidirectional communication protocol that can send the data from the client to the server or from the server to the client by reusing the established connection channel. The data is continuously pushed/transmitted into the same connection which is already open [22]. Unlike HTTP, where the client has to keep requesting updates (pooling), WebSockets allow the server to push updates to the client as soon as data changes.

4.4.2 Client-Side Technologies

4.4.2.1 React

React is a free and open-source front-end JavaScript library [12]. React is a library for helping developers build user interfaces as a tree of small pieces called components. This component is a mixture of HTML and JavaScript that capture all of the logic required to display a small section of a larger UI [13].

4.4.2.2 Javascript

JavaScript is a lightweight programming language commonly used by web developers to add dynamic interactions to web pages, applications, servers and even games. JavaScript's widespread applications in web, mobile app and game development make it a valuable language to learn [14].

4.4.2.3 Tailwind Css

Tailwind css is a low-level framework. Unlike other css frameworks like Bootstrap, Tailwind doesn't offer fully styled components like buttons, dropdowns and navbars. Instead, it offers a utility class which gives the option to create our own reusable components. Tailwind css provides a lot more flexibility and control over what the application looks like than other CSS frameworks [15].

4.4.3 Server Technologies

4.4.3.1 Node.js

Node.js is an open-source, cross-platform JavaScript runtime environment and library for running web applications outside the client's browser. Developers use Node.js to create server-side web applications and it is perfect for data-intensive applications since it uses an asynchronous, event-driven model. Node.js operates on a single-threaded event driven architecture, utilizing an event loop to handle multiple concurrent operations without blocking. Node.js's non blocking, event-driven architecture makes it ideal for real-time applications, web servers, APIs and more [16].

4.4.3.2 Fiware

FIWARE is an open source initiative that works towards building a set of standards to develop smart applications for different domains such as Smart Cities, Smart Ports, Smart Logistics, Smart Factories and others. FIWARE promotes a standard that describes how to collect, manage and publish context information, and additionally adds certain elements that allow exploiting collected data [17].
The FIWARE platform provides a rather simple yet powerful set of APIs that ease the development of smart applications in multiple vertical sectors [18].

4.4.3.3 MongoDB

MongoDB is a document database built on a horizontal scale-out architecture that uses a flexible schema for storing data. Instead of storing data in tables of rows or columns like SQL databases, each record in a MongoDB database is a document described in BSON, a binary representation of the data. Applications can then retrieve this information in a JSON format [19].

5 Work Artifacts

5.1. Requirement

5.1.1 Functional Requirements

No.	Requirement
1	The system will allow the administrator to approve new sign-ups
2	The system will allow users to select devices from a map
3	The system will allow users to choose which devices are shown on the map
4	The system will allow users to choose attributes for each device
5	The system will allow users to copy selected data
6	The system will allow users to mark an attribute as a favorite
7	The system will allow users to hide a device's attributes
8	The system will allow users to expand the information window
9	The system will allow users to minimize the information window
10	The system will allow users to add graphs of an attribute to the expanded window
11	The system will allow users to remove graphs from the expanded window
12	The system will allow users to view the device in 360 degrees
13	The system will allow users to close the selected device
14	The system will allow access to the site through mobile devices
15	The system will support dark mode functionality
16	The system will allow users to log in
17	The system will allow users to sign up

table 1. Functional Requirements

5.1.2 Non-Functional Requirements

No.	Requirement	type
1	The site should be easy to use and quick - 85% of users should be able to complete key tasks without assistance	Usability
2	The site should be easy to maintenance and update - Code modularity should support updates with minimal downtime (<5 minutes)	Maintainability
3	The site should be mobile friendly (WCAG 2.2, Web Content Accessibility Guidelines 2.2 - 2023) [21]	Adaptability
4	The site should be up to date	Maintainability
5	The site should be lightweight (e.g.,load time up to 3 seconds)	Efficiency, Performance
6	The site will save cookies	Usability
7	The site will work properly even if non of the devices are available	Fault tolerance
8	The system should support up to 5000 numbers of users connected at the same time without degradation in performance	Scalability
9	Only authorized users should be able to access the data - Multi-factor authentication (MFA) for all access points	Security
10	The system should comply with relevant data protection regulations (e.g., GDPR)	Compliance
11	The system should be designed to accommodate future enhancements with minimal changes - New modules or features can be integrated with no more than 10% modification of existing code	Extensibility
12	The system should provide interactive features that allow users to manipulate and visualize data in real-time - Response time for interactive queries should be < 1 second	Interactivity
13	The system's components should be modular and reusable in other projects - At least 80% of system components should be reusable in similar projects	Modularity

table 2. Non-Functional Requirements

5.2. Use Case

In the Use case diagram there are 2 players:

a) **Unapproved user** - player which can only sign up for the website and needs to wait until gets approved by the supervising professor.

b) **Approved user** - player which can sign in and do all the regular operations in the website including choosing the device from the map, clicking and watching all of its attributes. Moreover the player can copy the attribute, pin it in the list of attributes in the device details screen and generate a graph. The player can generate a graph with the last 100 samples or with specific arguments and can specify the graph type he wants to generate.

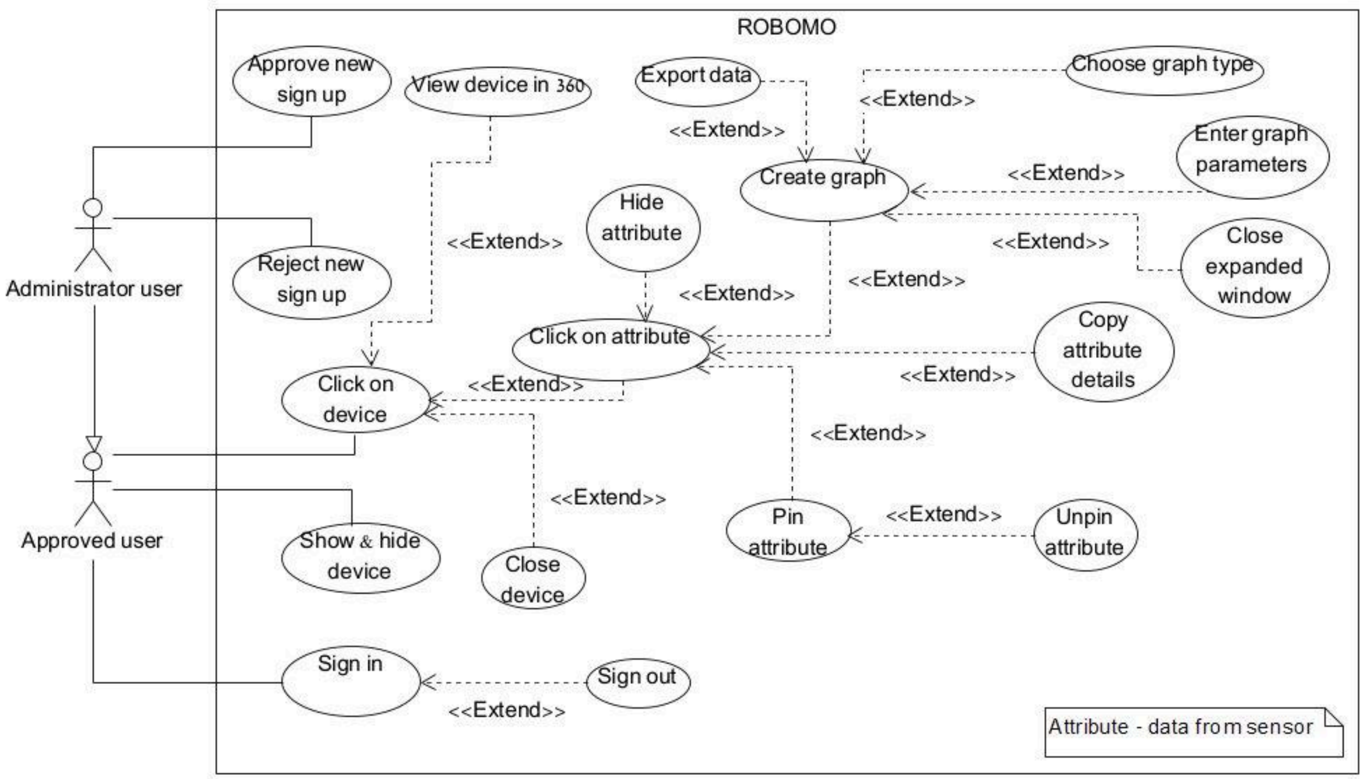


figure 3. UseCase diagram

5.3. Activity Diagram

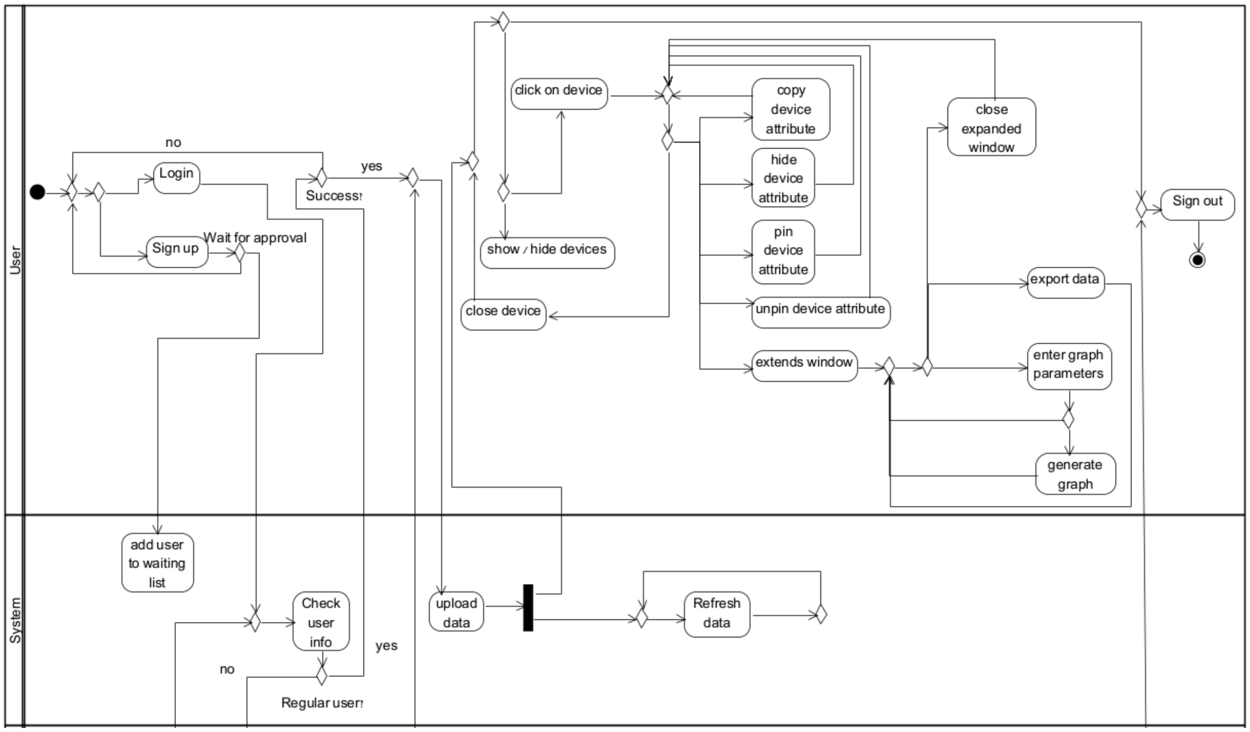


figure 4a. Activity diagram - User and System

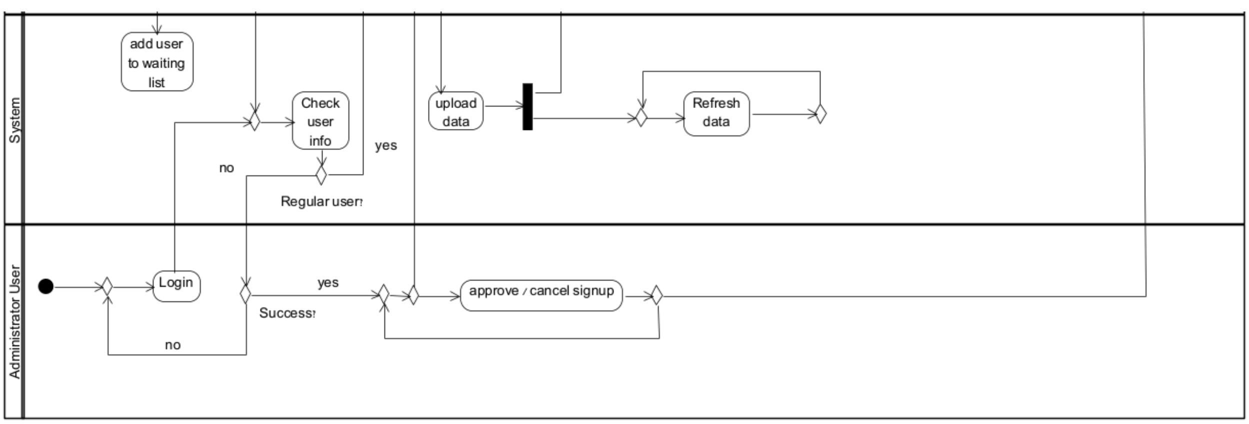


figure 4b. Activity diagram - Administrator user and System

5.4. Website Screens

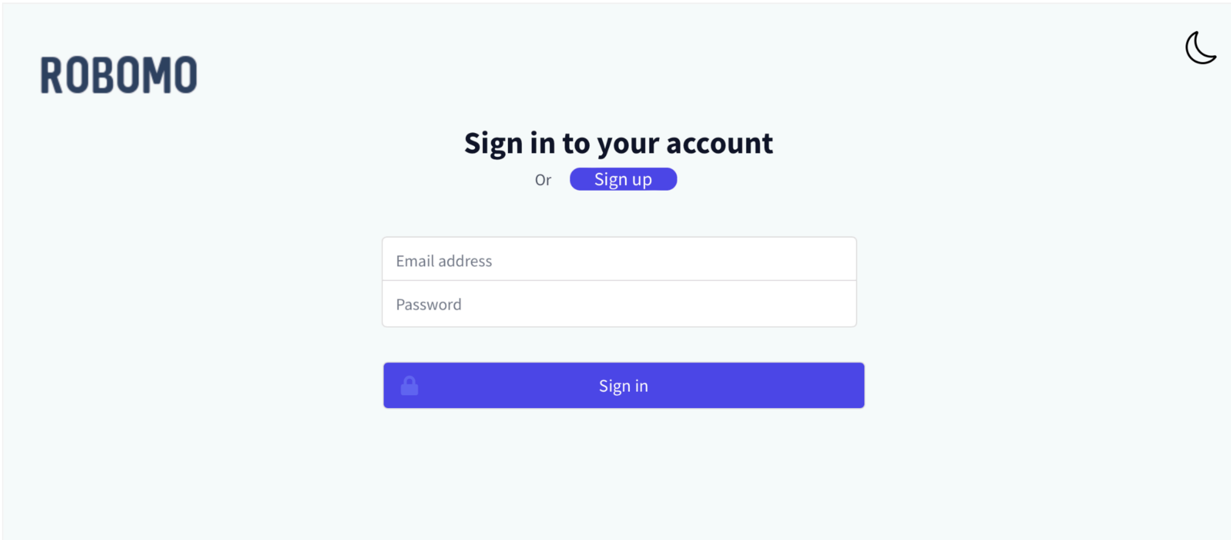


figure 5. Login screen

figure 6. Home screen

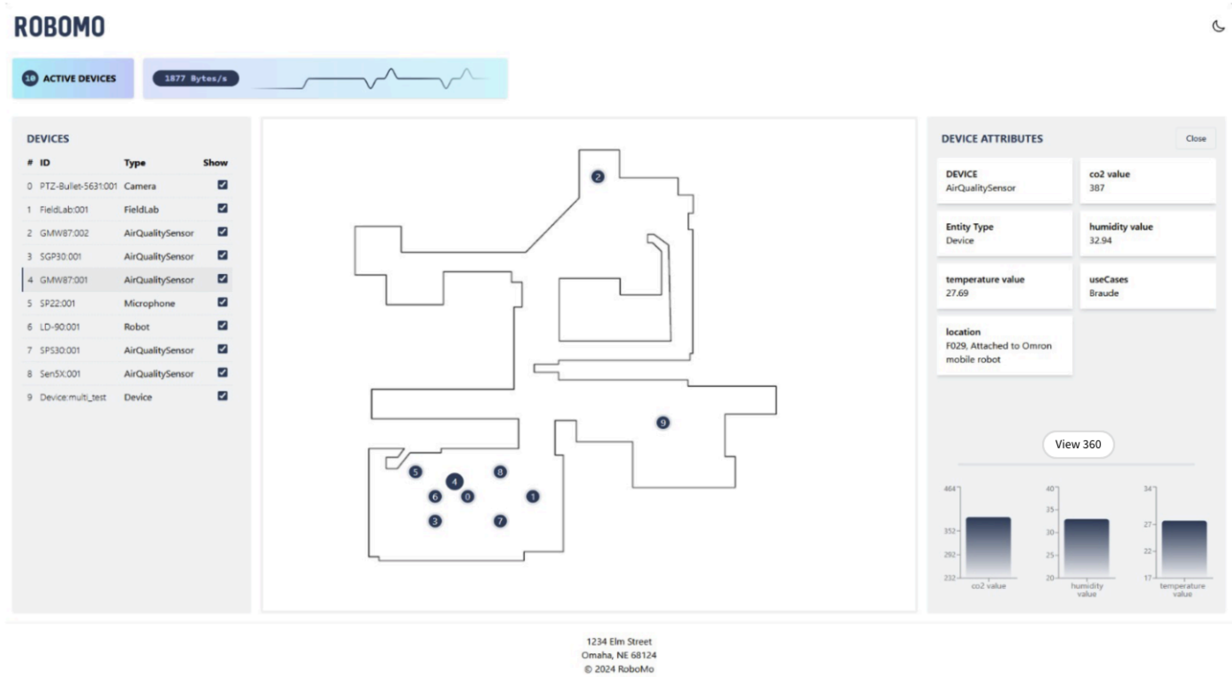


figure 7. Selecting device screen

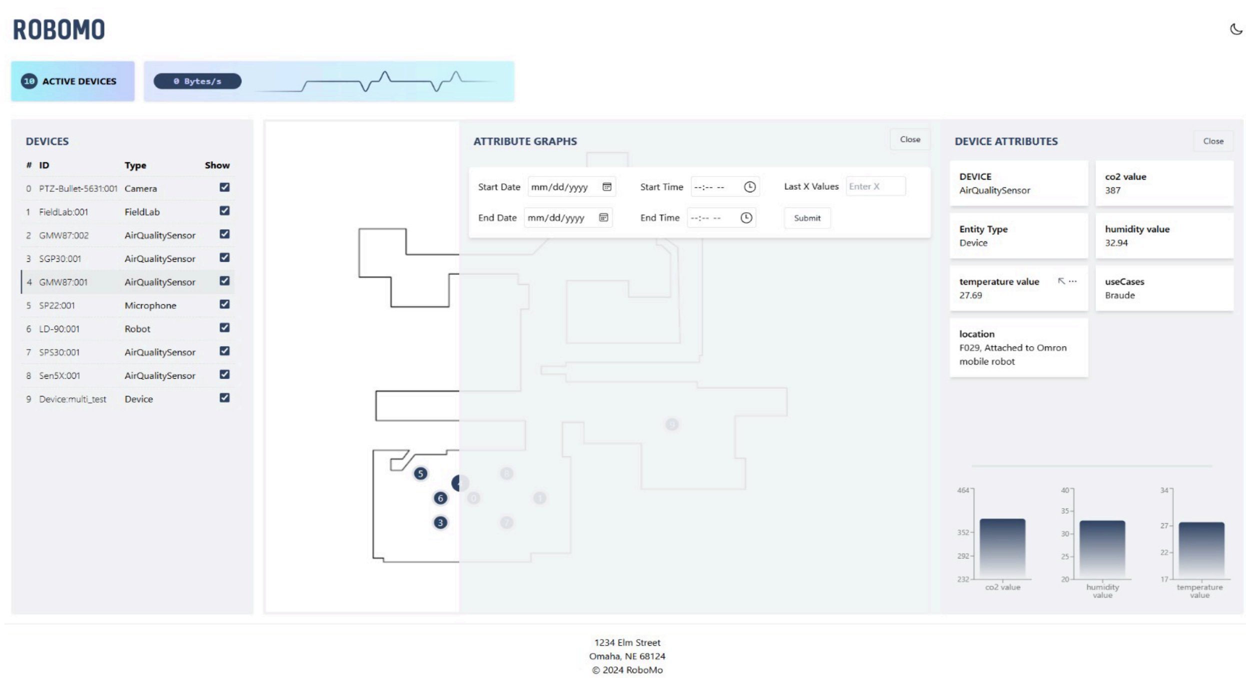


figure 8. Expanded window

figure 9. Generating graph from expanded screen

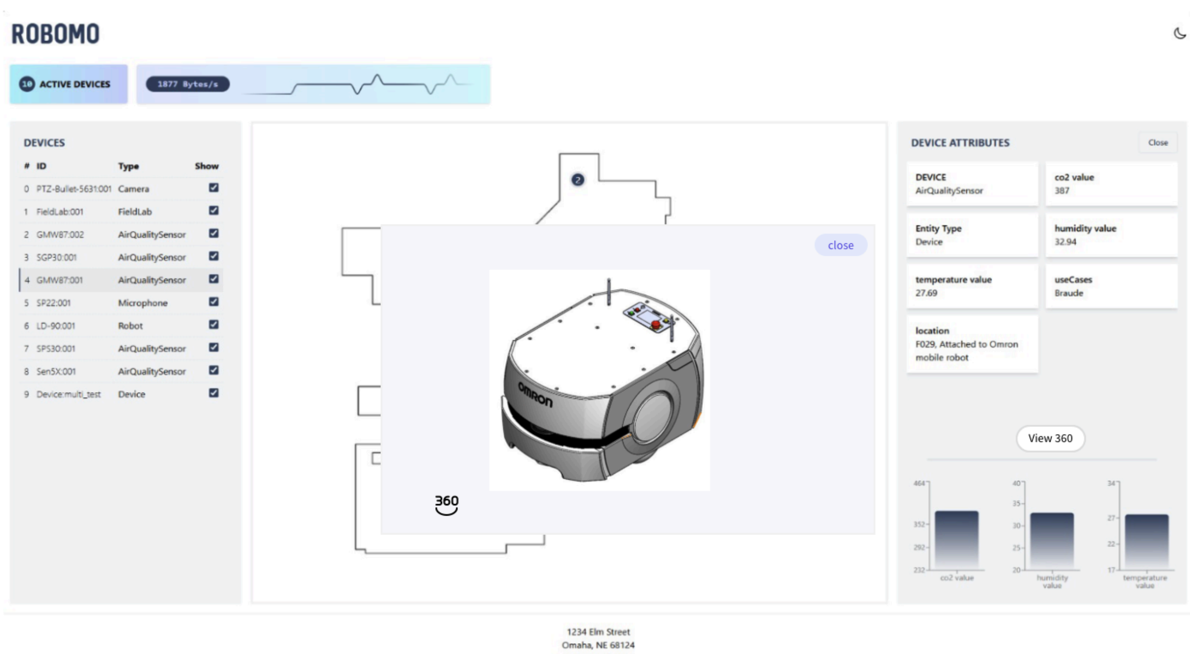


figure 10. 3D view

6. Expected Achievements

a) **Seamless API integration:** Establishing a reliable connection to external APIs, despite the need for VPN use, ensuring uninterrupted data access.

b) **Demonstration of Robust IoT and Data Monitoring Capabilities:** Showcasing the system's ability to monitor and process live data from robotics and sensors across various environments.

c) **Effective Use of Agile Methodology:** Successfully applying Agile principles, including regular feedback loops, sprints, and iterative improvements, to guide project development.

d) **User-Friendly Interface:** The product will provide an intuitive interface that allows users to easily monitor, manage and visualize the data collected from connected devices.

e) **Scalable Architecture:** The system will be designed to support increasing numbers of devices, ensuring scalability without compromising performance.

f) **High Data Accuracy:** The website will guarantee that all sensor data is transmitted and displayed with high precision, providing users with reliable and accurate information.

6.1. Challenges

a) **Communication Barriers:** Team members are located in different countries, which makes communication a challenge. Misunderstandings or delays in communication can hinder progress.

b) **Adaptivation:** Changing from traditional CSS to Tailwind CSS to create a more adaptive website for smartphone access.

c) **Connection stability**: Relying on a VPN to access APIs from a different country leads to slower, unstable connections and complex configurations which impact the project's performance.

6.2. Success Criteria

a) Data from the API is retrieved and displayed within an acceptable time frame (e.g., 2 seconds).

b) The website performs well under various loads, handling multiple simultaneous users without significant performance degradation (e.g., 20 users).

c) The system has a high uptime percentage, consistently maintaining an availability of 95.0%, ensuring reliable access and minimal downtime for users.

d) The website is easy to maintain and update, with clear documentation and well-structured code (e.g., IEEE 828-2012) [20].

e) 85% of users should be able to complete key tasks without assistance

6.3. Evaluation

The evaluation of the project will be conducted using the System Usability Scale (SUS) questionnaire, accompanied by open-ended questions to gather detailed feedback for improvements. This approach will help assess the overall usability of the system while identifying areas for enhancement. After the SUS questionnaire, the project will be demonstrated to lab workers for further review. Additionally, it will be presented to students who are enrolled in relevant courses, ensuring that feedback is collected from individuals with varying levels of experience in the subject. This comprehensive evaluation will guide the final adjustments to the system.

7. Testing Plan

7.1. Scope

This scope encompasses automated unit and integration testing of React components and JavaScript modules as well as manual testing of user interfaces and workflow.

7.2. Objectives

a) Ensure the system meets all specified functional and non-functional requirements.

b) Identify and document any bugs, issues, or areas for improvement.

c) Validate the system's reliability and stability over extended periods of use.

7.3. Testing Approach

Our testing approach will prioritize automation and continuous integration to insure code quality and system reliability throughout the development process. We'll implement a robust unit testing framework using Jest for our React components and JavaScript modules, aiming for high test coverage. Furthermore, we will conduct comprehensive tests on all interactive elements in order to ensure a seamless and intuitive experience.

7.4. Test Cases

#	TestID	Precondition	Expected Result	Description
1	SuccessLogin	The user approved by administrator, the user's data exists in the database	The user is logged in and sees the main screen	1)User inserts his userName. 2)User inserts his password. 3)User clicks the “Login” button.

2	FailLoginNotApproved	The user is not approved by the administrator	An error message is displayed - “Wait for approval”	1)User inserts his userName. 2)User inserts his password. 3)User clicks the “Login” button.
3	FailLoginWrongPassword	The user inserts wrong password	An error message is displayed - “Wrong password”	1)User inserts his userName. 2)User inserts his password. 3)User clicks the “Login” button.
4	SuccessSignIn	The user’s data doesn’t exist in the database	An message is displayed - “Complete, wait for approval”	1)User inserts userName. 2)User inserts password. 3)User clicks the “Sign-in” button.
5	FailSignInUserNameTaken	The user’s data exists in the database	An error message is displayed - “UserName already exist”	1)User inserts userName. 2)User inserts password. 3)User clicks the “Sign-in” button.
6	FailSignInMissingData	The user didn’t fill in all necessary information.	An error message is displayed - “Please insert username and password”	1)User inserts password. 2)User clicks the “Sign-in” button.
7	ChoosingDevice	1)The user login successfully. 2)The data was fetched successfully.	Device attribute screen will update with the data	User clicks on some device in the map
8	SortDisplay	1)The user login successfully. 2)The data was fetched successfully.	The map will update and show the chosen devices.	User clicks on the devices in Device screen he wants to show on map
9	SuccessApproveUser	1)The user login successfully. 2)The type of user is “Administrator”. 3)The data of pending users is fetched successfully.	An message is displayed = “Approved user successfully”	Administrator clicks on “approve” button of some pending user in the Pending screen
10	SuccessDenyUser	1)The user login successfully. 2)The type of user is “Administrator”. 3)The data of pending users is fetched successfully.	An message is displayed = “deny user successfully”	Administrator clicks on “deny” button of some pending user in the Pending screen

11	SuccessCreateGraph	1)The user login successfully. 2)Data was fetched successfully	An updated graph will appear in the extended window	1)User clicks in the device attribute screen on attribute. 2)User clicks on the expanded window button. 3)User inserts a range of time and date. 4)User clicks on the “Submit” button.
12	SuccessOperational	1)The user login successfully. 2)Data wasn’t fetched successfully	The website will be empty but operational.	The website is running and operational although there’s no incoming data from the server.

table 3. Test cases

8. AI Tools and Prompts

During our work on part a, we used AI tools such as: ChatGPT, Claude and Copilot. We used these different AI tools in a variety of areas such as checking syntax and spelling, conducting literature introduction research as well as in the practical part to check on various technologies and methods.

8.1. Prompts Used

Prompt 1: “We develop an integrated internet of things software system that monitors robotics data, using various sensors that are on the robotics and visualizes that data. Please suggest an appropriate name for such a site”.
Answer: ChatGPT suggested different kinds of names for such systems including: “RoboSense”, “MonitorBot”, “IoT Sphere”, “RoboMo” and such more. Eventually we decided to choose the name “RoboMo” which stands for: Robotics Monitoring.

Prompt 2: “We develop an integrated internet of things software system that monitors robotics data, using various sensors that are on the robotics and visualizes that data. Please provide relevant literature about IoT,IIoT, robotics and monitoring data”.
Answer: ChatGPT and Claude provided us with different types of articles that connected to the subjects we mentioned. This helped us in writing the literature review and finding more suitable terms that are also connected to our project.

Prompt 3: “We want to use better server-client communication in our React project, please list the best ways to achieve this”.
Answer: ChatGPT provided us several approaches that we can choose depending on our use case, project requirements and scalability needs. Among these approaches chatGPT introduced WebSocket, which we chose eventually, and explained the way it works. ChatGPT explained that it enables full-duplex communication between the client and the server and that it is ideal for real-time, bidirectional communication and live updates and explained in detail about the advantages of this approach.

9. References

[1] Demertzi, V., Demertzis, S., & Demertzis, K. (2023). An Overview of Privacy Dimensions on the Industrial Internet of Things (IIoT). Algorithms, 16(8), 378.

[2] Fuller, A., Fan, Z., Day, C., & Barlow, C. (2020). Digital twin: Enabling technologies, challenges and open research. IEEE access, 8, 108952-108971.

[3] Grieco, L. A., Rizzo, A., Colucci, S., Sicari, S., Piro, G., Di Paola, D., & Boggia, G. (2014). IoT-aided robotics applications: Technological implications, target domains and open issues. Computer Communications, 54, 32-47.

[4] Kamilaris, A., & Botteghi, N. (2020). The penetration of Internet of Things in robotics: Towards a web of robotic things. Journal of ambient intelligence and smart environments, 12(6), 491-512.

[5] Khan, A., Aziz, S., Bashir, M., & Khan, M. U. (2020, March). IoT and wireless sensor network based autonomous farming robot. In 2020 international conference on emerging trends in smart technologies (ICETST) (pp. 1-5). IEEE.

[6] Krishnamurthi, R., Kumar, A., Gopinathan, D., Nayyar, A., & Qureshi, B. (2020). An overview of IoT sensor data processing, fusion, and analysis techniques. Sensors, 20(21), 6076.

[7] Li, P., & Liu, X. (2019, July). Common sensors in industrial robots: A review. In Journal of Physics: Conference Series(Vol. 1267, No. 1, p. 012036). IOP Publishing.

[8] Nayyar, A., Puri, V., Nguyen, N. G., & Le, D. N. (2018). Smart surveillance robot for real-time monitoring and control system in environment and industrial applications. In Information Systems Design and Intelligent Applications: Proceedings of Fourth International Conference INDIA 2017 (pp. 229-243). Springer Singapore.

[9] Ray, P. P. (2016). Internet of robotic things: concept, technologies, and challenges. IEEE access, 4, 9489-9500.

[10] Singh, M., Fuenmayor, E., Hinchy, E. P., Qiao, Y., Murray, N., & Devine, D. (2021). Digital twin: Origin to future. Applied System Innovation, 4(2), 36.

[11] Vermesan, O., Bröring, A., Tragos, E., Serrano, M., Bacciu, D., Chessa, S., ... & Bahr, R. (2022). Internet of robotic things–converging sensing/actuating, hyperconnectivity, artificial intelligence and IoT platforms. In Cognitive hyperconnected digital transformation (pp. 97-155). River Publishers.

[12] React

[13] What React Is and Why It Matters [Book]

[14] What Is JavaScript: A Beginner's Guide to the Basics of JS

[15] Tailwind CSS: What It Is, Why Use It & Examples

[16] What Is Node.js? A Complete Guide for Developers

[17] FIWARE, the standard that the IoT needs – TM Forum

[18] What is FIWARE?

[19] Why Use MongoDB And When To Use It?

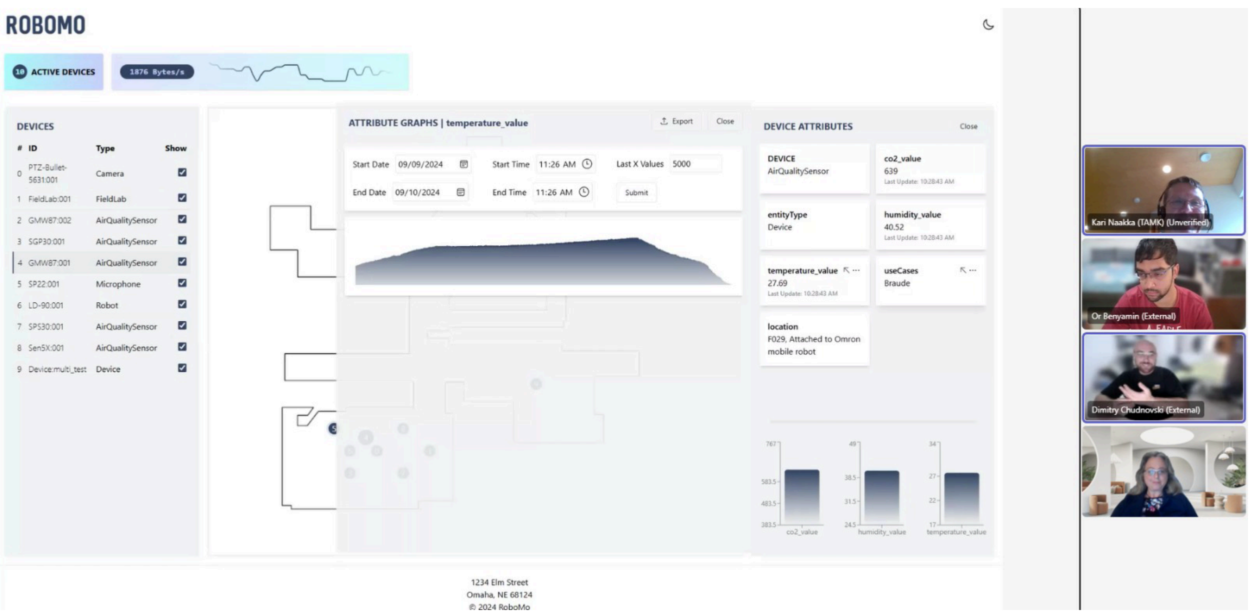
[20] IEEE SA - IEEE 828-2012

[21] Web Content Accessibility Guidelines (WCAG) 2.2

[22] What is web socket and how it is different from the HTTP? - GeeksforGeeks

10. Appendices

10.1. Meeting Capture



picture 1. 4th meeting with the customer in Teams