   
**Technological University Dublin**   
**Blanchardstown Campus**

Bachelor of Engineering in Computer Engineering

H3024 Project 2

*Project Title:*

Remote machine monitoring and tracking device

*Author:*

Erikas Valiukevicius

B00141111

TU807

*Supervisor:*

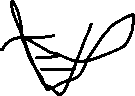
Mark Deegan

*Date:*

18/2/2024

# Declaration

The material contained in this assignment is the author’s original work, except where work quoted is duly acknowledged in the text. No aspect of this assignment has been previously submitted in any other unit or course.



Signed: Date: 18/2/2024

# Abstract

# Acknowledgements

I would like to thank Mark Deegan, Ivan smyth and Fergus maughan for helping me in the development of this project and assisting in my learning throughout my time in their care.

I would also like to thank all my other lecturers whose classes aided me in the understanding and development of the necessary skills to be able to perform at my best for this project.

# Table of Contents

Declaration ii

Abstract iii

Acknowledgements iv

Table of Contents v

Abbreviations, Symbols, Acronyms vi

List of Figure Captions vii

List of Tables viii

Product Specifications 1

Executive Summary 2

Chapter 1 Introduction 3

1.1 The Aim: 3

1.2 Motivation: 3

1.3 Layout of Report: 4

Chapter 2 Background Research 5

Chapter 3 Project Design 6

3.1 **Theory and reasoning for the choice.** 6

3.2 **Hardware design** 6

**Communication Interfaces:** 7

3.3 Software Design 10

3.4 Software Flow Chart 11

**3.5Alternative design** 11

**3.6 Discussion** 11

Chapter 4 Project Construction and Testing 13

4.1 Construction 13

4.2 Testing Procedure 13

4.3 Software Testing 13

4.4 Fault Tree 13

4.5 Results 13

Chapter 5 Safety and Ethical Considerations of the Project. 14

5.1 Health and Safety Issues: 14

5.2 Ethical Considerations: 15

Chapter 6 Conclusions and Recommendations 16

References 17

Appendix 18

Timetable 18

# Abbreviations, Symbols, Acronyms

1. **ESP-32**: Espressif Systems' ESP32, a low-power system-on-chip microcontroller widely used in IoT applications.
2. **IoT**: Internet of Things, a network of interconnected devices that communicate and exchange data.
3. **LoRa**: Long Range, a wireless communication technology for long-range, low-power IoT applications.
4. **NB-IoT**: Narrowband Internet of Things, a cellular communication standard optimized for IoT devices.
5. **Wi-Fi**: Wireless Fidelity, a wireless networking technology based on IEEE 802.11 standards.
6. **Bluetooth**: A wireless technology standard for short-range communication between devices.
7. **GPS**: Global Positioning System, a satellite-based navigation system providing location and time information.
8. **MQTT**: Message Queuing Telemetry Transport, a lightweight messaging protocol for IoT applications.
9. **Azure**: Microsoft Azure, a cloud computing platform for building, deploying, and managing applications and services.
10. **RoHS**: Restriction of Hazardous Substances, a directive restricting the use of certain hazardous materials in electrical and electronic equipment.
11. **CE**: Conformité Européenne, a certification mark indicating conformity with health, safety, and environmental protection standards for products sold within the European Economic Area.
12. **UL**: Underwriters Laboratories, a safety certification company providing safety-related certification, validation, testing, inspection, and auditing services.
13. **GDPR**: General Data Protection Regulation, a European Union regulation on data protection and privacy for all individuals within the EU and the European Economic Area.
14. **ISO**: International Organization for Standardization, an international standard-setting body responsible for developing and publishing standards.
15. **HIPAA**: Health Insurance Portability and Accountability Act, a US legislation ensuring data privacy and security provisions for safeguarding medical information.
16. **CCPA**: California Consumer Privacy Act, a state statute intended to enhance privacy rights and consumer protection for residents of California, United States.
17. **ISO 9001**: International Organization for Standardization's standard for quality management systems, demonstrating a commitment to meeting customer requirements and continuous improvement.
18. **ISO 14001**: International Organization for Standardization's standard for environmental management systems, focusing on minimizing environmental impact and improving resource efficiency.
19. **GD**: Global Distribution, a supplier of electronic components and technologies.
20. **RS**: RS Components, a global supplier of electronic components and solutions.
21. **Farnell**: Farnell Ireland, a distributor of electronic components and related products.

# List of Figure Captions

[An automated list similar to the table of contents]

# List of Tables

[An automated list similar to the table of contents]

# Product Specifications

(Consult with you supervisor as to the type of list of specification (if any) needed here.)

# Executive Summary

(This section summarizes the entire report in such a way that the reader can gain all the salient information contained in the main report without having to read the entire report)

# Chapter 1 Introduction

# 1.1 The Aim:

This project's main goal is to create a device that monitors and tracks machines remotely. It's meant to help prevent issues like misuse, abuse, and breakdowns by giving real-time monitoring and logging abilities. By keeping an eye on things like temperature, pressure, speed, impacts, and location, it'll help businesses deal with maintenance needs early, use machines better, and save money.

Here are the main things we're focusing on:

1. **Hardware Setup**: We're putting together the ESP32-S3 microcontroller with other hardware pieces, and we'll be using communication protocols like I2C and SPI.
2. **Wireless Connection**: The device will use Wi-Fi and Bluetooth to connect wirelessly. It'll talk to a personal hotspot so you can manage it from far away.
3. **MQTT Setup**: We're setting up MQTT on the ESP32-S3 to help with lightweight messaging and connecting to other parts of the device. We'll be using Mosquitto for smooth communication.
4. **Cloud Services**: We're putting a virtual system on Microsoft Azure, using Ubuntu Linux. Then, we'll set up Docker containers for important services like NodeRED, Mosquitto, InfluxDB, Grafana, and Portainer to make a flexible and scalable backend.
5. **Docker and Containerization**: We'll use Docker and Docker Compose to organize things neatly, so we can easily manage all those different services.
6. **NodeRED**: We're going to use NodeRED to make it easy to connect different parts of the system and move data between them.
7. **Testing**: We'll test everything thoroughly to make sure it all works well together. We want to be sure it's reliable, stable, and safe.Top of Form

# 1.2 Motivation:

The motivation behind this project is:

1. Business cost reduction: business cost of unexpected repairs due to mismanagement of machinery would be greatly reduced.
2. By implementing a remote monitoring and tracking solution, businesses can gain better insights into machinery health, identify potential issues early on, and take proactive measures to prevent breakdowns and minimize downtime.

Moreover, there is a growing emphasis on environmental sustainability and reducing carbon emissions.

Machinery breakdowns not only incur financial costs but also contribute to increased resource consumption and environmental impact.

By implementing an effective monitoring system, businesses can optimize machinery usage, reduce the need for repairs and replacements, and ultimately minimize their carbon footprint.

# 1.3 Layout of Report:

The purpose of this report is to provide a comprehensive overview of the entire project life cycle.

It begins with an introduction outlining the aims and motivation of the project.

the report then delves into background research and explores relevant technologies, platforms, and concepts necessary to understand the scope of the project.  
Next, the report describes the design phase of the project, including hardware and software design considerations, construction procedures, and testing. methods.

In addition, attention is paid to security and ethical aspects to ensure that the project complies with relevant standards and regulations and ensures both human safety and data protection.  
Finally, the report presents more important results, conclusions and recommendations based on the results of the project.

It provides an overview of the effectiveness of the developed solution, its potential impacts on businesses and the environment, and areas that should be investigated and improved in the future.  
Overall, this report is a comprehensive project document and provides valuable information about the development of the solution.

# Chapter 2 Background Research

This section provides an overview of the technologies and concepts relevant to the development of the remote machine monitoring and tracking device.

* 1. **MIT App Developer**:

The MIT App Inventor is a web application, integrated development environment originally provided by Google. It allows users to create software applications for the Android operating system using a visual drag-and-drop interface.

the platform has a wide range of pre-built components and features that a user can incorporate into their application. These vary from buttons,text boxes, images, sensors, media playback, and much more. MIT app inventor allows users to focus on functionality as opposed to coding syntax and other intricacies.

* 1. **Node-RED**:

Node-RED is a flow-based development tool for visual programming developed by IBM. It is built on Node.js and provides a browser-based editor that makes it easy to wire together devices, APIs, and online services.

* 1. **MQTT (Message Queuing Telemetry Transport)**:

MQTT is a lightweight messaging protocol ideal for IoT applications. It follows a publish-subscribe messaging pattern, making it efficient for communication between devices and servers.

* 1. **Azure**:

Microsoft Azure is a cloud computing platform that offers a wide range of services, including computing, analytics, storage, and networking. Azure provides robust infrastructure and platform services to support IoT solutions, including data storage, analytics, and machine learning capabilities.

* 1. **ESP32**:

The ESP32 is a low-power system-on-chip microcontroller with integrated Wi-Fi and Bluetooth capabilities. It is widely used in IoT applications due to its versatility, low cost, and power efficiency.

* 1. **Adafruit Ultimate GPS v3**:

The Adafruit Ultimate GPS v3 module is a GPS module based on the MTK3339 chipset. It provides accurate positioning information with support for multiple satellites, including GPS, GLONASS, and Galileo.

# Chapter 3 Project Design

## 3.1 **Theory and reasoning for the choice.**

The project aims to address the issue of machinery mistreatment and breakdowns by providing a tracking and monitoring solution.

The choice of technologies is based on their suitability for achieving this goal.

1. MIT App Inventor and Node-RED were selected for their ease of use in developing mobile and IoT applications.
2. MQTT was chosen as the messaging protocol for its efficient communication over networks.
3. Azure IoT Hub offers cloud-based data storage, making it suitable for storing sensor data. The ESP32 microcontroller was selected for its versatility and built-in Wi-Fi and Bluetooth connectivity.
4. The Adafruit Ultimate GPS module provides accurate positioning data for tracking machinery locations.

## 3.2 **Hardware design**

1. ESP32 microcontroller:

• The ESP32 microcontroller acts as the central unit of the tracking device, responsible for interfacing with the sensors, connecting to the MQTT broker and managing data.

• It offers built-in Wi-Fi capabilities, making it suitable for connecting to local networks and communicating with cloud services.

2. Sensors:

• Temperature Sensors: Used to monitor the temperature of the machine, detect overheating or anomalies.

• Vibration Sensors: Detect mechanical vibrations and shocks and signal possible malfunctions or shocks to the machine.

• Adafruit Ultimate GPS- Module: Provides accurate location tracking. - and geographic information that enables real-time tracking and location of machines.

3. Power source:

• Power sources depend on the installation environment and the power requirements of the monitoring device.

• Alternatives are rechargeable batteries, solar panels for remote use or direct electrical connections for permanent installations.

## **Communication Interfaces:**

* A wireless network is a computer network that uses wireless data connections between network nodes. Wireless networking allows homes, telecommunications networks and business installations to avoid the costly process of introducing cables into a building, or as a connection between various equipment locations.
* Wireless networking for this project was a personal hotspot for the setup, as the esp32-s3 already has built in 2.4 GHz Wi-Fi and Bluetooth module installed

### I2C

I2C ( inter integrated Circuit) is a synchronous, multi-master, multi-slave communication protocol.

You can connect :

multiple slaves to one master: for example, your ESP32 reads from a sensor using I2C and writes the sensor readings in an I2C OLED display.

multiple masters controlling the same slave: for example, two ESP32 boards writing data to the same I2C OLED display.

We used I2C over for example the serial port as serial ports are asynchronous.

Meaning?

This means that there is no clock data transmitted over the serial port, devices need a clock data to synchronise over to communicate correctly with each other.

I2C communication happens over 2 communication lines SDA and SCL which are both active low and as such require pull up resistors.

### SPI

Serial Peripheral Interface is a de facto standard for synchronous serial communication, used primarily in embedded systems for short distance wired communication between integrated circuits.

These devices are organized into a master and slave configuration, in which the master has control over the slaves and the slaves receive instruction from the master.

There are four signals required to implement SPI communication.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **SIGNAL** |  | **DESCRIPTION** | | MOSI |  | Data: Master Out — Slave In | | MISO |  | Data: Master In — Slave Out | | SCLK |  | Serial Clock | | CS |  | Chip Select | |

The clock signal is used to synch the data received and transmitted between devices.

### TCP/ IP

Transmission Control Protocol (TCP) is a communications standard that enables application programs and computing devices to exchange messages over a network. It is designed to send packets across the internet and ensure the successful delivery of data and messages over networks.

TCP organizes data so that it can be transmitted between a server and a client. It guarantees the integrity of the data being communicated over a network. Before it transmits data, TCP establishes a connection between a source and its destination, which it ensures remains live until communication begins. It then breaks large amounts of data into smaller packets, while ensuring data integrity is in place throughout the process.

### MQTT

MQTT is a standard messaging protocol for the Internet of Things (IoT).

It is designed as an extremely lightweight publish/subscribe messaging transport that is ideal for connecting remote devices with a small code footprint and minimal network bandwidth. MQTT today is used in a wide variety of industries, such as automotive, manufacturing, telecommunications, oil and gas, etc.

In this project the MQTT was called Mosquitto which is an open-source message broker. It is a very lightweight system for carrying messages.

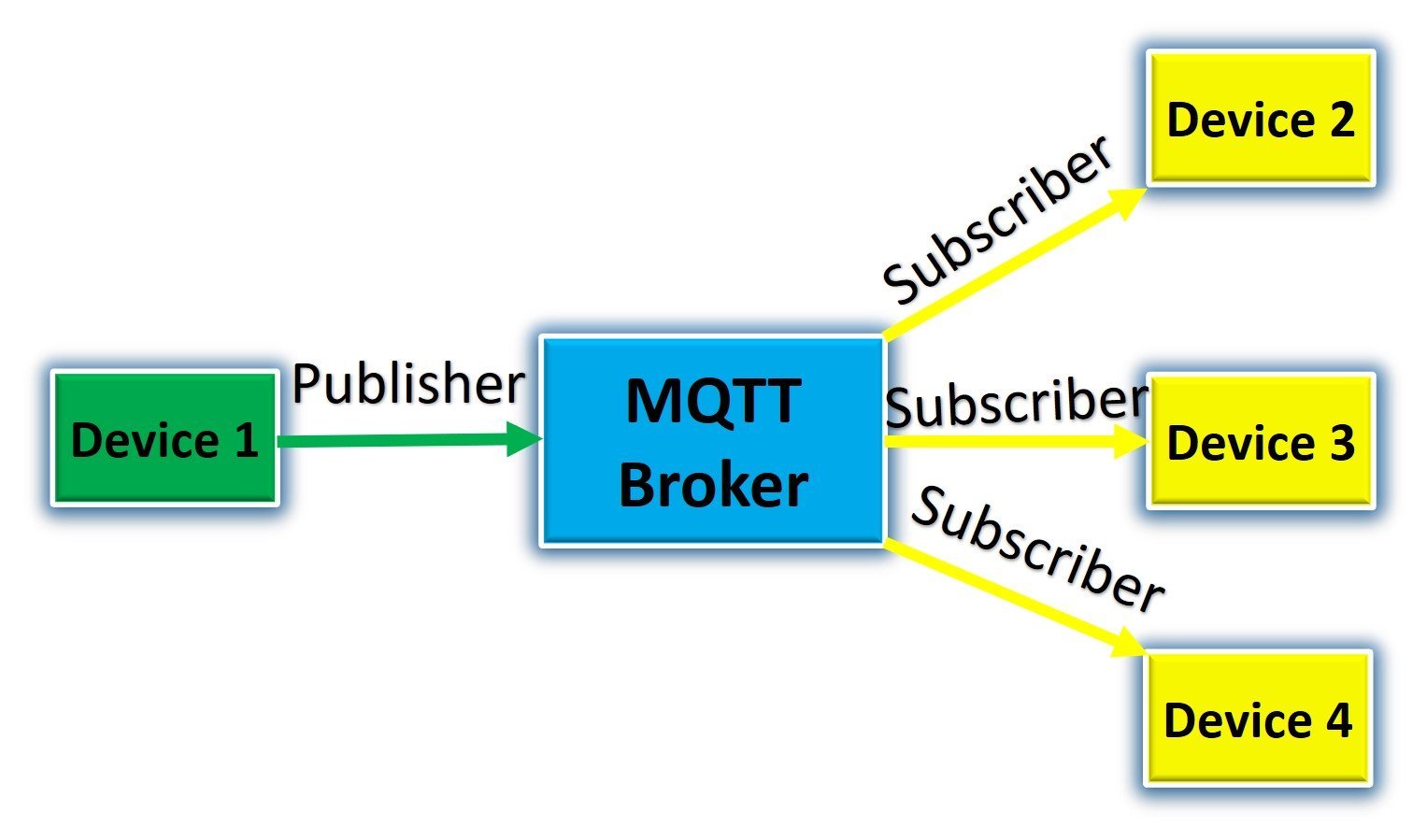


Figure 3 mqtt broker

### SSH

The Secure Shell (SSH) protocol is a method for securely sending commands to a computer over an unsecured network. SSH uses cryptography to authenticate and encrypt connections between devices. SSH also allows for tunnelling, or port forwarding, which is when data packets can cross networks that they would not otherwise be able to cross.

A screenshot of a computer program

Description automatically generated

Figure 4 ssh client

It's a bit like a secret passage for computer information. People often use SSH to control faraway computers, manage tech stuff, or share files without worrying about anyone spying.

Remote encrypted connections: SSH sets up a connection between a user's device and a faraway machine, often a server. It uses encryption to scramble the data that traverses the connection. An intercepting party would only find something like static — random data that means nothing unless it is decrypted. (SSH uses encryption methods that make decryption prohibitively difficult for outsiders.)

The ability to tunnel: In networking, tunneling is a method for moving packets across a network using a protocol or path they would not ordinarily be able to use. Tunneling works by wrapping data packets\* with additional information — called headers — to change their destination. SSH tunnels use a technique called port forwarding to send packets from one machine to another.

1. **Enclosure:**
   * The hardware enclosure provides protection for electronic components against environmental factors such as moisture, dust, and physical damage.
   * Enclosures should be designed to accommodate all components securely and allow for easy access to sensors and connectors for maintenance and troubleshooting.
2. **Environmental Considerations:**
   * Temperature, humidity, and exposure to elements may impact the performance and longevity of hardware components.
   * Selecting components rated for industrial or outdoor use ensures reliability and resilience in harsh environments.

A computer screen shot of a computer

Description automatically generated

## 3.3 Software Design

**Backend Infrastructure:**

**Cloud Database:** Using Azure IoT Hub as a cloud-based database solution to store machine data. The database is designed to accommodate a variety of data types, including sensor readings, machine IDs, timestamps. MQTT Broker: MQTT broker was implemented to facilitate communication between the ESP32 controller and the cloud database. MQTT is chosen for its publish-subscribe communication protocol, which is suitable for IoT applications.  
**Data:** Development of back-end scripts and services to process incoming sensor data.

**Frontend Infrastructure:**

**Mobile** **App:**

 Design a mobile application using MIT App Inventor to provide users with an intuitive interface to access machine data. The application allows users to view real-time sensor readings, track machine location on a map.  
Integration and Interoperability:  
**ESP32 Firmware:**

Development of ESP32 microcontroller firmware to collect sensor data from various sensors such as temperature sensors, pressure sensors, vibration sensors and Adafruit Ultimate GPS module. The firmware communicates with the MQTT broker using the MQTT protocol for perfect integration with the backend infrastructure.

## 3.4 Software Flow Chart

## **3.5Alternative design**

Alternative design may involve using different microcontrollers, communication protocols or project-based cloud platforms. requirements and limitations. For example, alternative microcontrollers such as Arduino boards or Raspberry Pi could be considered, and different cloud platforms such as AWS IoT or Google Cloud IoT could be used.

## **3.6 Discussion**

The design choices for this project aim to respond to specific tracking. and monitoring needs. needs machine control in various industrial environments. However, there are several aspects and potential challenges to discuss:

**Technology applicability:**

Selected technologies, including MIT App Inventor, Node-RED, MQTT, Azure IoT Hub, ESP32, and Adafruit Ultimate GPS module, were selected based on their ease of use. integration and functionality. Although these technologies offer great opportunities, integrating them seamlessly into a single system during testing can be a challenge.

**Scalability:**

One of the main considerations is the scalability of the system to accommodate many machines and sensors.

As the number of monitored machines increases, the system must be able to efficiently handle the additional data flow.

This requires careful optimization of data transfer protocols to ensure scalability without compromising performance.

**Data security and privacy:**

The project involves the collection and transfer of sensitive machine data, which raises data security and privacy concerns.

Strong encryption mechanisms must be implemented to protect data during transmission and storage.

 In addition, access control measures must be implemented to prevent unauthorized access to the system and ensure compliance with data protection regulations.

**Reliability and fault tolerance:** System reliability is critical for real-time monitoring and tracking of machine status.

To ensure reliability, fail-safe mechanisms should be implemented to handle communication failures, sensor failures, and other unexpected events.

This may mean consolidating data paths, sensor calibration routines, and error detection and correction mechanisms.

**User interface** **and experience**: usability for a mobile application and dashboard interface plays an important role in user adoption and satisfaction.

The user interface must be intuitive, visually appealing and responsive so that users can easily access relevant machine information.

User feedback and usability testing are essential to improving the UI and overall user experience.

**Future** **Improvements:** While the current project scope is focused on basic machine monitoring and tracking features, there is potential for future improvements and extensions.

This may include incorporating machine learning algorithms for predictive maintenance, integrating additional sensor types for deeper monitoring, and exploring advanced data visualization techniques to gain deeper insights into machine performance.

A close-up of several white squares

Description automatically generated

# Chapter 4 Project Construction and Testing

## Construction

## Testing Procedure

## Software Testing

## Fault Tree

## Results

# Chapter 5 Safety and Ethical Considerations of the Project.

Ensuring the security and ethical integrity of the remote machine monitoring and tracking device is critical to its successful implementation and operation. This chapter addresses health and safety issues related to both the hardware and software aspects of the project, as well as the ethical considerations involved in producing multiple units of the device.

# 5.1 Health and Safety Issues:

**Hardware:**

1. **Electrical Safety:**

Proper insulation and grounding of electrical components are essential to prevent electrical shocks such as electric shocks or short circuits. All components must meet relevant safety standards such as CE or UL certifications to ensure compliance with electrical safety regulations.

1. **Mechanical Safety:**

The equipment enclosure must be designed to minimize the risk of physical injury or damage to users and machinery. Rounded edges, secure mounting, and anti-knock design features enhance safety during installation and use.

1. **Environmental Safety:**

Components selected for equipment design must be evaluated for intended environmental conditions, including temperature, humidity, and exposure to dust or moisture. Compliance with environmental directives such as RoHS (Restriction of Hazardous Substances) ensures the use of environmentally friendly materials and processes.

**Software:**

1. **Data Privacy and Security:**

The implementation of encryption protocols and secure data transfer mechanisms ensure the security of sensitive data collected by the monitoring device. Compliance with data protection rules such as the General Data Protection Regulation (GDPR) protects user privacy and prevents unauthorized access to confidential information.

1. **System Reliability:**

Robust software engineering and rigorous testing procedures reduce the risk of system failures or malfunctions, reducing potential malfunctions or security risks.

1. **User Training and Documentation:**

Clear instructions and manuals educate users on the safe handling and use of the tracking device, reducing the risk of user error or accidents.

# 5.2 Ethical Considerations:

1. **Fair Labor Practices:**

Ensuring fair wages, working conditions and labor rights for people involved in the production and assembly of tracking devices promotes ethical manufacturing practices and supports worker welfare.

1. **Environmental Sustainability:**

The use of environmentally friendly materials, energy-efficient components and recyclable packaging minimizes the environmental impact of the production and disposal of the device. Compliance with international standards such as ISO 14001 environmental management systems demonstrates commitment to sustainability.

1. **Transparency and Accountability:**

Providing open information about device features, limitations, and potential risks increases trust and accountability among stakeholders, including end users, manufacturers, and regulatory agencies.

**Adherence to Relevant Standards:**

1. **Electrical Safety Standards:**

Compliance with standards such as IEC 61010 ensures the safety of electrical equipment and users.

1. **Data Privacy Regulations:**

Adherence to regulations such as HIPAA (Health Insurance Portability and Accountability Act) and CCPA (California Consumer Privacy Act) protects sensitive user data and ensures legal compliance.

1. **Product Quality Standards:**

Certification under ISO 9001 for quality management systems demonstrates adherence to international quality standards and commitment to product excellence.

# Chapter 6 Conclusions and Recommendations

[You should draw conclusions from the work you have done on your project over the year and any recommendations for improvements etc if work were continuing on it.]

# References

1. Espressif Systems. (n.d.). ESP32 Technical Reference Manual. Retrieved from <https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf>
2. Adafruit Industries. (n.d.). Ultimate GPS v3. Retrieved from <https://learn.adafruit.com/adafruit-ultimate-gps/downloads>
3. MIT App Inventor. (n.d.). About MIT App Inventor. Retrieved from <https://appinventor.mit.edu/explore/library>
4. Node-RED. (n.d.). About Node-RED. Retrieved from <https://nodered.org/about/>
5. MQTT. (n.d.). MQTT Version 3.1.1. Retrieved from <http://www.steves-internet-guide.com/mqtt/>
6. Microsoft Azure. (n.d.). Azure Documentation. Retrieved from <https://docs.microsoft.com/en-us/azure/>
7. LoRa Alliance. (n.d.). What is LoRaWAN? Retrieved from <https://lora-alliance.org/about-lorawan>
8. IEEE Standards Association. (n.d.). IEEE Standards for Industrial Automation. Retrieved from <https://ieeexplore.ieee.org/browse/standards/reading-room/page>
9. Occupational Safety and Health Administration (OSHA). (n.d.). Machine Guarding. Retrieved from <https://www.osha.gov/machine-guarding>
10. Espressif Systems. (n.d.). ESP32 Technical Reference Manual. Retrieved from <https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf>

# Appendix

Specification sheets for components, software code, AutoCAD drawing, Etc

**NO PADDING**

Padding is where you add in data sheets for components which are standard and do not need to be included such as a common transistor type. It can also be the inclusion of 20 pages of a data sheet when only 4 of them are relevant.

Quantity does not necessarily mean quality.

**Circuit Diagrams from CadStar**

* Include proper block diagrams and circuit diagrams. **NO SCREEN DUMPS FROM ECAD**.
* Add in the printed diagrams in your report at the end or print to a PDF file and include it in the soft copy of your report.

# Timetable

Include here the timetable you used for Project 1 and your timetable for Project 2 in tabular form (from Excel).