# Introduction

[insert introduction]

## Problem Definition

The demographic landscape is undergoing a transformative shift globally, with an increasing aging population requiring specialised attention and care. One prominent challenge faced by the elderly and their caregivers is the need for real-time monitoring of environmental conditions, particularly temperature variations, within living spaces. [insert reference] In many instances, maintaining a comfortable and safe environment becomes a critical factor in ensuring the health and well-being of elderly individuals, especially when their ability to regulate their surroundings may be compromised. [insert reference]

Research shows that traditional monitoring systems often fall short in providing a seamless, unobtrusive solution tailored to the unique needs of the elderly. [insert reference] Therefore, this project aims to develop an innovative IoT-based solution utilising temperature sensors and data analytics to create a responsive environment. By doing so, the project aims to enhance the quality of life for elderly individuals while additionally supporting caregivers in their responsibilities. [insert reference]

This project will delve into the intricacies of hardware and software integration, exploring the capabilities of Arduino-based technology, and importance of cloud-based services for efficient data storage and analytics. It is anticipated that this project will not only address a specific problem but also contribute to the broader discourse on the responsible and inclusive use of technology in the realm of elderly care.

# Project Requirements

The project requirements require a range of hardware, software, and cloud services to achieve the objectives outlined in the introduction.

## Hardware Requirements

* **Arduino Uno WIFI R2 Board:** This board will provide seamless integration of IoT sensors, providing the necessary computational power for data processing and transmission.
* **Grove Seeed Studio Temperature Sensor v1.3:** This is a precise temperature sensor compatible with the Arduino board that will ensure accurate and real-time environmental monitoring within living spaces.
* **Grove-LCD RGB Backlight Display v5.0:** This will provide a clear visual representation of temperature variations. The visual feedback enhances user comprehension and interaction.
* **Power Supply:** In development mode the Arduino is powered by USB connected to a PC. In run mode the Arduino is powered by a 9V battery. Considering the potential battery usage, the battery ensures continuous operation and data capture for a prolonged amount of time.

## Software/ Cloud Requirements

* **Arduino Code:**
  1. **VS Code IDE:** For programming the Arduino board, this software platform simplifies the firmware development and code development.
  2. **Programming Languages**: Arduino Sketch (C++) These languages facilitate effective communication between the Arduino board and connected sensors.
  3. **Code Design**: This will include configurations to connect to the internet using DNS, receive accurate time updates through NTP, communicate with cloud services using MQTT with Mosquitto, send data updates to ThingSpeak, and seamlessly obtain network settings through DHCP.
* **ThingSpeak Platform with HTTP:** The project requires a database and data visualisation.ThingSpeak has been chosen as its user-friendly interface facilitates efficient data storage, management, analytics, and visualisation. A channel will be set up in ThingSpeak to receive and analyse data sent from the Arduino. Seamless integration with this cloud platform will be fundamental to the projects data management and analytical capabilities. The analytical features of ThingSpeak will provide insightful analysis of temperature trends over time. ThingSpeak will function as the central hub for data storage and analytics, offering real-time insights through its user interface.
* **MQTT Mosquitto Platform:** MQTT was used to provide machine to machine (M2M) connectivity with the Arduino acting as the MQTT publisher. The MQTT broker at test.mosquitto.org was used. C:\Program Files\mosquitto>mosquitto\_sub -h test.mosquitto.org -t warmth-checker
* **Network Time Protocol (NTP):** To maintain real time on Arduino to aid in temperature updates.
* **Domain Name Server (DNS):** A name resolution service to convert Ip address to readable network address.
* **Dynamic Host Configuration Protocol (DHCP):** to get an Ip address from router for Arduino to operate on Wi-Fi network.

## Additional

* **Internet Connectivity**: A reliable internet connection is necessary for efficient data transmission between the Arduino device and ThingSpeak as well as the MQTT broker. A stable connection is paramount for the project's real-time responsiveness.
* **Responsive Environment:** To enhance the project's user interaction and environmental adaptability, a responsive environment that dynamically adjusts to temperature variations will be needed to activate the LED display to communicate changes.

# Circuit Design

# System Design

# Project Testing phase

# Data Analytics

# Legal & Ethical Evaluation

## Legal Considerations

Data Privacy:

* Requirement: Ensure compliance with data privacy regulations to protect the personal information collected through the IoT device.
* Implementation: Implement encryption protocols and anonymize data to adhere to regulations such as the General Data Protection Regulation (GDPR) (European Commission, 2016).

Intellectual Property:

* Requirement: Verify that the project does not infringe on existing patents or intellectual property rights.
* Implementation: Conduct thorough patent searches and ensure that all code and designs adhere to open-source or appropriately licensed frameworks.

## Ethical Considerations

Informed Consent:

* Requirement: Ensure users are adequately informed about the purpose and functionality of the IoT device.
* Implementation: Provide clear documentation and consent forms for users, emphasizing transparency in data collection and usage.

Accessibility:

* Requirement: Design the IoT device to be accessible to all users, including those with disabilities.
* Implementation: Incorporate features such as voice commands or alternative interfaces to enhance accessibility (World Health Organization, 2011)

## Commercial & Economic Context

Cost-Effectiveness:

* Requirement: Evaluate the economic feasibility and cost-effectiveness of the IoT solution.
* Implementation: Conduct a cost-benefit analysis considering both initial implementation costs and long-term maintenance expenses.

Market Viability:

* Requirement: Assess the potential market demand and viability of the developed IoT solution.
* Implementation: Conduct market research to identify user needs and preferences, ensuring alignment with the commercial market.

## Sustainability, Equality, Diversity, and Inclusion (EDI) Issues

Environmental Impact:

* Requirement: Evaluate the environmental impact of the IoT device, considering its lifecycle.
* Implementation: Choose materials and manufacturing processes with lower environmental impact and design for energy efficiency.

Inclusivity:

* Requirement: Ensure the IoT device caters to a diverse user base, considering cultural and demographic variations.
* Implementation: Design interfaces and user experiences that are culturally sensitive and inclusive (World Health Organization, 2013).

# conclusion

# source code

# reference

# appendix