

1 Name of Use Case

Name of the Use Case	Integrated IoT Platform for Elderly Care
Version No.	V0.1
Submission Date	01/12/2024
Team Members (with student IDs)	Thiago Baldassarri Levin 334838 Barbara Valdete Correia da Cunha 287436 Francesca Fornasier 338688 Asia Ferri 331439

2 Scope and Objectives of Function

Scope and Objectives of Use Case	
Scope	The proposed IoT aims to integrate safety features and health monitoring for elderly individuals.
Objective(s)	The system will monitor vital signs and personal safety, providing real-time alerts to caregivers as needed.
Domain(s)	Elderly care, smart healthcare
Stakeholder(s)	Private caregivers, Family Members, Nursing Homes
Short description	<p>The IoT platform integrates wearable sensors to provide real-time safety and health monitoring for elderly individuals. It includes accelerometers for fall detection, health sensors for tracking vitals, an emergency button, and a caregiver notification system. Using MQTT and REST APIs, the platform ensures efficient communication between devices. Key features include:</p> <ul style="list-style-type: none">• Fall detection and alerts.• Vital signs monitoring.• Scalable integration through a Catalog.• Multi-channel notifications via apps.

- Telegram Bot missing. Maybe telegram bot instead of Raspberry Pi with display?

- Provide a clear explanation of the post-processing services. Remember that they must be "complex" enough (not just if-else or for loop). For each of the services provide a high level overview of the processing algorithm, and specify its inputs and outputs (with its corresponding communication protocols)

- Missing description Thingspeak

To retrieve historical data to be shown, this data should be retrieved from the Thingspeak adaptor using REST, and not directly through thingspeak

early mornings) or irregular activity during nighttime, **prompting the system to suggest targeted caregiver interventions.** How will this "output" be sent? REST, MQTT?

- **ThinkSpeak Adaptor:** bridges the platform to the ThinkSpeak cloud for data storage and visualization. It receives environmental and health data via MQTT and uploads it to ThinkSpeak using REST APIs, enabling historical data analysis and integration with third-party tools.

- **Freeboard:** acts as a real-time and historical data visualization dashboard, providing intuitive graphs and **metrics** for health readings and alerts. This year we will see Nodered instead of Freeboard in the course, as explained during lectures. Do you still want to use Freeboard?

- **Raspberry Pi with display:** At the scheduled times, caregivers will receive reminders to give the medications. Delivers real-time notifications about health alerts, ensuring caregivers stay informed about critical issues as they arise. Implementation of other modules to add informations about caregivers or patients.

- **Catalog:** a centralized registry for all devices, services, and configurations. It dynamically provides MQTT topics, REST endpoints, and configuration settings for all components. This ensures that new devices or services can be added without requiring changes to the platform's source code, allowing the system to be scalable.

The system's data flow begins with sensors collecting real-time information on motion, temperature, and health metrics. The device Connector processes this data and publishes it to the Message Broker via MQTT. Data is simultaneously uploaded to the ThinkSpeak cloud for historical storage and visualization, which Freeboard accesses for real-time monitoring. Alerts and critical information are also transmitted to a Raspberry Pi and are shown on the display.

This IoT platform will offer a robust solution for elderly safety and health monitoring. Its real-time alert capabilities, dynamic scalability, and user-friendly interfaces ensure a seamless experience for caregivers and users. The platform's focus on monitoring and notifications makes it an essential tool for ensuring the safety and well-being of elderly individuals.

5 Desired Hardware components (only among those we can provide)

Device Name	Quantity	Needed for...

If you want to "show" some processed data, remember that the UI is just for displaying data... any calculation must be done in a separate microservice. If that is the case, you might add one new microservice for this processing of the data to be shown...

Maybe Telegram Bot? Who will send the alerts about health? Timer Control? Clearly specify inputs and outputs of this microservice (with the corresponding communication protocol)