A LoRaWAN Multi-Network Server Application for Smart Cold Chain Tracking in Remote Areas

Alex Fabiano Garcia[1][a], Wanderley Lopes de Souza[1][b] and Luís Ferreira Pires[2][c]

[1] *Ubiquitous Computing Group, Graduate Program in Computer Science, Computing Department, Federal University of São Carlos, P.O. Box 676, São Carlos, Brazil*

[2] *Semantics, Cybersecurity & Services Group, Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, P.O. Box 217, Enschede, The Netherlands*

Keywords: Internet of Things, LoRaWAN, Transport, GPS-free, Multilateration, Cold Chain, Perishable Goods.

Abstract: This paper describes the design and implementation of a multi-network server application for smartly tracking cold chain based on the LoRaWAN technology. The system aims to solve logistics challenges in remote areas by using IoT technologies to monitor and manage the conditions of perishable goods during transportation, ensuring their quality and safety. The proposed solution encompasses various sensor types and integrates multiple network providers to improve coverage, aiming to support decision-makers in public-private partnerships when addressing social issues in outlying regions. The study compares the simulation of antenna coverage with the effective distance from the gateway that receives the signal. Field tests in the Netherlands demonstrate the system's effectiveness in real-world scenarios, showing features such as GPS-free geolocation by multilateration, long-range communication, and the potential for applying our solution in other domains beyond cold chain logistics.

1 INTRODUCTION

The term Internet of Things (IoT), coined by Kevin Ashton in 1999 (Ashton et al., 2009), refers to a network of pervasive devices embedded with sensors, software, and other technologies to collect and exchange data. IoT significantly transformed various application domains, such as smart homes, industrial automation, healthcare, and environmental monitoring, by enabling unprecedented levels of efficiency, automation, and data-driven decision-making (Atzori et al., 2010).

One critical application domain of IoT is cold chain logistics, where IoT solutions help ensure the integrity and safety of temperature-sensitive products such as pharmaceuticals, food, and chemicals. IoT enhances cold chain management by providing real-time monitoring and data analytics, which in turn helps maintain product quality and comply with regulatory standards (Badia-Melis et al., 2018; Aung and Chang, 2014).

The cold chain logistics sector faces significant challenges, particularly in remote areas where maintaining the required temperature conditions for sensitive products is arduous due to energy restrictions and lack of network connectivity (Badia-Melis et al., 2018). For example, the antivenom distribution in the Amazon region (Fan and Monteiro, 2018) is a concrete case in which cellular connectivity is often sporadic or non-existent, hindering real-time monitoring and data transmission, and leading to delays in identifying and addressing temperature oscillations. The Internet penetration in Brazilian rural areas is around 34%, compared to 65% of the urban regions, mainly due to the high investment per covered inhabitant and revenue uncertainty from Mobile Network Operators (MNO) (Cavalcante et al., 2021). Addressing these challenges requires innovative solutions that operate efficiently under energy constraints and provide reliable communication in areas with limited network infrastructure.

Designed for low-power wide-area communication, LoRaWAN (Long Range Wide Area Network) is a network protocol that enables long-range data transmission with minimal energy consumption, making it appropriate for IoT applications in locations with limited access to power and connectivity (Centenaro

[a] https://orcid.org/0000-0002-6202-1849
[b] https://orcid.org/0000-0001-8645-4393
[c] https://orcid.org/0000-0001-7432-7653

Garcia, A. F., Lopes de Souza, W. and Ferreira Pires, L.
A LoRaWAN Multi-Network Server Application for Smart Cold Chain Tracking in Remote Areas.
DOI: 10.5220/0013211600003929
In *Proceedings of the 27th International Conference on Enterprise Information Systems (ICEIS 2025) - Volume 1*, pages 1049-1060
ISBN: 978-989-758-749-8; ISSN: 2184-4992

et al., 2016). Its robust network architecture operating in a license-free band allows for scalable deployment, accommodating thousands of connected devices and facilitating comprehensive environmental monitoring in remote cold chain logistics (Vangelista et al., 2015). Additionally, the total daily cost for an increasing number of devices is lower than that of cellular protocol alternatives due to their subscription costs and private infrastructure (Frangoudis et al., 2021).

In our research, we identified gaps in related designs that employ LoRaWAN in cold chain logistics scenarios, developed a multi-network server application that aims to fill some of these breaches in remote locations, and deployed and evaluated a prototype of this application in the field. In our multi-network approach, in which several LoRaWAN network servers are applied, we sought to enhance network coverage by leveraging existing infrastructure provided by the public LoRaWAN Network Operator (LoRa Alliance, 2024) and private networks. Moreover, we applied the LoRaWAN geolocation capabilities without using Global Positioning Systems (GPS).

This paper is further structured as follows: Section 2 details our research methodology; Section 3 describes the most relevant related work; Section 4 presents an overview of the leading technologies employed in this work; Section 5 describes the design of the system developed in this study; Section 6 presents tests and validation of our system; Section 7 concludes the paper by discussing our contributions, the remaining challenges, and some topics for future work.

2 RESEARCH METHODOLOGY

In this research we followed the Design Science Methodology (DSM) (Wieringa, 2014). Figure 1 shows the framework used, which considers two main activities: (a) designing an artefact and (b) empirically investigating its effects in a problem context. By building a multi-network LoRaWAN application, we aim to improve cold chain tracking while satisfying long-range communication, condition breach detection, and GPS-free geolocation. Our application aims to help governments establish public-private partnerships (PPP), investing in private LoRaWAN deployments while taking advantage of existing infrastructure from different network providers, enabling continuous cold chain tracking in remote areas with a cost-effective technology (Frangoudis et al., 2021).

Our artefact has been conceived in a Design Cycle consisting of three tasks: 1. *Problem Investigation*: to capture stakeholder objectives and identify

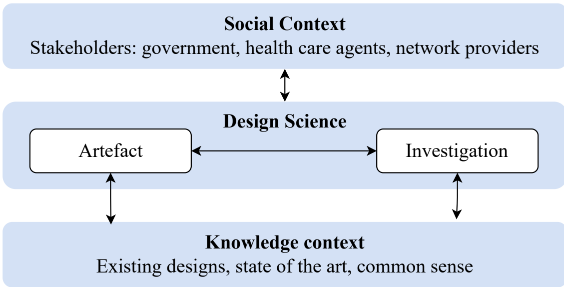


Figure 1: A Framework for Design Science (adapted from (Wieringa, 2014)).

the challenges faced in tracking the cold chain in remote regions; 2. *Treatment Design*: to design the LoRaWAN application to improve cold chain tracking in remote areas with sparse network coverage; and 3. *Treatment Validation*: to evaluate the effects caused by our LoRaWAN application in the context of cold chain tracking. The Treatment Validation task could trigger a new iteration of the Design Cycle, as shown in Figure 2. We used an agile development approach to gradually design, implement, and evaluate our application throughout the design cycle.

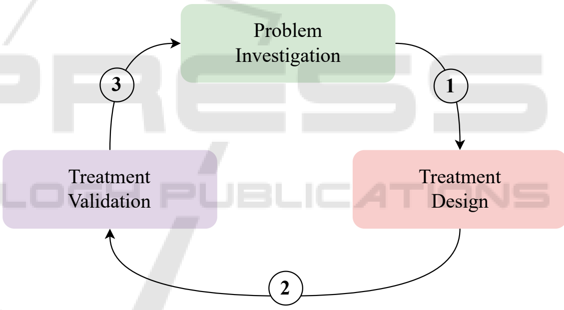


Figure 2: The Design Cycle tasks.

According to (Wieringa, 2014), the Design Cycle is part of a more comprehensive flow called the Engineering Cycle, which also encompasses the additional Treatment Implementation and Treatment Evaluation tasks to validate the application of the artefact in a real-world scenario and use and evaluate it with the interested parties. Since we only validated our application in experimental settings, these tasks were out of the scope of our research project.

In the Problem Investigation task, we first conducted a Systematic Literature Review (SLR) (Garcia and de Souza, 2023) of IoT applications and technologies for cold chain tracking. In this SLR, we identified the lack of network infrastructure that supports the distribution of cold chain goods, especially due to the low Return on Investment (RoI) for telecom companies, considering the population density in remote areas. Moreover, we also learned how low-power wide-area network (LPWAN) technologies such as LoRaWAN offer appropriate capabilities in resource-constrained contexts. Next, we conducted a comprehensive literature review of this technology to identify its potential role as part of a solution. Although many applications employ this technology, they do not exploit some aspects, especially the hybrid networking capability.

In the Treatment Design task, we developed a multi-network LoRaWAN application named Cold Chain Tracking (CCT), which includes hardware prototypes, the communication layer, and a web application. The Treatment Validation comprises a Single-Case Mechanism and a Scale-Up Experiment (Wieringa, 2014). We first tested our application in the field with different routes and transport modes, validating the fulfilment of the requirements defined before. Finally, we scaled up to more realistic conditions via load tests to assess the robustness of the application.

3 RELATED WORK

Our previous study (Garcia and de Souza, 2023) highlights several IoT applications and technologies for cold chain vaccine tracking through an SLR. Among them, in (Lawrence et al., 2018) LoRaWAN and IoT are applied to support the health sector in Kikwit, Democratic Republic of the Congo. The authors implement a mesh network in a real-world application, demonstrating the suitability of the protocol in resource-limited contexts and areas with poor infrastructure. They mention the difficulties in finding suitable high-altitude locations to install gateways in an ad-hoc manner and the lack of financial support at national or regional levels to expedite their deployment.

In (Zinas et al., 2017), the authors report on a protocol and application based on the LoRaWAN technology for cattle tracking, implemented and tested in Pogoniani, Greece. The device prototype includes sensors, a GPS module, a LoRa transceiver, and a microcontroller enclosed in a box and placed in the cows' bell collar. During their experiments, the authors could send data over LoRaWAN to a distance of 6 km.

Focusing on maintaining optimal temperature conditions, in (Enriko et al., 2022) an IoT-based approach to monitoring vaccine quality during distribution in Indonesia is presented. The approach integrates various IoT devices and sensors, ensuring continuous monitoring and immediate alerts in case of deviations from the required conditions. Due to the network coverage, the proposed architecture employs LoRaWAN only for at-rest nodes while using cellular technologies for in-transit devices.

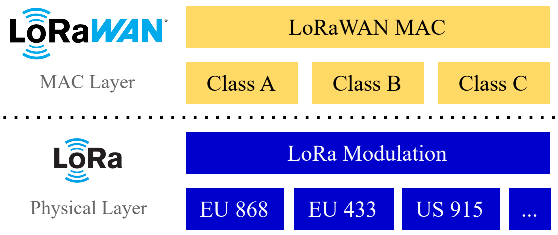


Figure 3: LoRa and LoRaWAN network protocol layers.

In (Bose et al., 2022), a LoRaWAN-based vaccine cold storage monitoring system is proposed that comprises an end node with a GPS module, LoRaWAN gateway, network server, and a web-based user application interface. The authors tested their design with a stationary gateway placed at a high altitude in Bangalore, India. Their presented results indicate that the protocol is noise-resistant for long-range communications in urban areas.

Our paper addresses some of the gaps from the related works by enabling a multi-LoRaWAN network to enhance the coverage, showing a method for antenna placement rather than ad-hoc attempts, and using the cheaper LoRaWAN GPS-free geolocation by multilateration (Fargas and Petersen, 2017).

4 IoT-RELATED TECHNOLOGIES

This section introduces some aspects of LoRa (Long Range), LoRaWAN, and MQTT (Message Queuing Telemetry Transport), which are the most relevant IoT-related technologies we used to develop our application.

4.1 LoRa and LoRaWAN

LoRa and LoRaWAN complement each other to define a Low-Power Wide Area Networks (LPWAN) protocol. Figure 3 shows that LoRa refers to the physical layer, which applies a wireless modulation technique based on Chirp Spread Spectrum (CSS) that enables long-range communication with minimal power consumption. This modulation allows for robust data transmission over several kilometres and operates on the license-free sub-Gigahertz bands, such as 915 MHz, 868 MHz, and 433 MHz, depending on the region (Augustin et al., 2016).

On its turn, LoRaWAN is the Media Access Control (MAC) layer protocol that sits on top of LoRa. It defines the communication protocol and system ar-

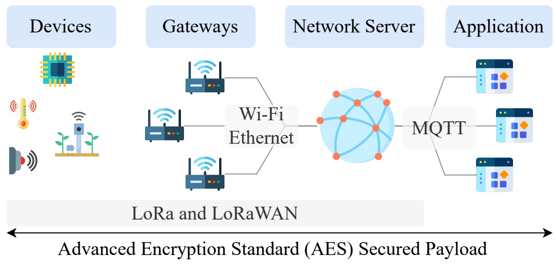


Figure 4: LoRaWAN architecture layers and technologies.

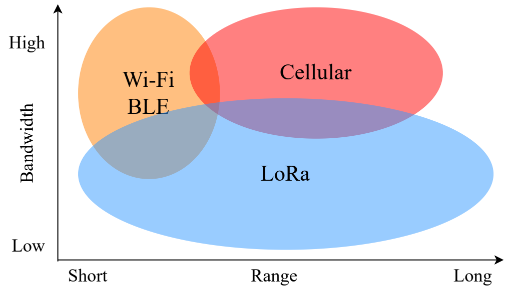


Figure 5: LoRaWAN bandwidth and range compared to other wireless networks (adapted from The Things Network[1]).

chitecture for orchestrating the network operation, including how devices connect and transmit data, ensuring security and scalability (Cattani et al., 2017). Since the protocol operates on regional license-free bands, governments often regulate the duty cycle of radio devices, which indicates the fraction of time a resource is allowed to be busy.

LoRaWAN defines three classes of devices to address different application needs regarding communication latency and energy efficiency: Class A devices have the lowest power consumption, are designed for battery-operated sensors, and allow two short receive windows following each uplink transmission; Class B devices add scheduled receive slots, using a synchronised beacon from the gateway to open additional receive windows at set times, balancing power consumption with more frequent downlink opportunities; Class C devices are nearly always listening for downlinks, offering minimal latency at the cost of higher power consumption, suitable for applications where immediate response is critical like control systems or actuators (Augustin et al., 2016; Cattani et al., 2017).

4.2 LoRaWAN Architecture

Figure 4 illustrates a typical LoRaWAN architecture, which is composed of end devices, gateways, network servers, and applications. The end devices contain sensors or actuators and send and receive LoRa-modulated wireless messages to the gateways, which are connected to the Network Server through a backhaul such as Wi-Fi, Ethernet, cellular networks or satellite. The Network Server manages all the LoRaWAN network components and interfaces the data exchange with the application where the business logic resides, usually through MQTT brokers.

The Network Service includes the software responsible for handling device activation in the network, the Join Server, which allows the end nodes to connect using either Over-The-Air-Activation (OTAA) or Activation By Personalisation (ABP) (Augustin et al., 2016). The former is a preferred method, where the device sends a join request to the network server, which then responds with session keys used to encrypt future communications, allowing dynamic rejoining and better security. In contrast, the latter involves pre-configuring devices with network parameters like session keys and device addresses before deployment, enabling immediate communication but lacking the dynamic security and flexibility of OTAA.

Enterprise Network Servers such as The Things Stack (TTS)[1] offer plug-and-play capabilities for custom devices and gateways to operate alongside each other. Telecom providers usually include gateways as part of the solution, with features built on top of that such as GPS-free geolocation via triangulation, trilateration, and multilateration (Fargas and Petersen, 2017). In contrast, open-source alternatives like ChirpStack[2] enable complete private network operations attending the LoRaWAN specifications.

4.3 Bandwidth and Range

Figure 5 shows that wireless networks such as Wi-Fi and Bluetooth (including its Low Energy version BLE) have short range but high bandwidth, making them ideal for video and voice package transmissions. Cellular technologies are intended for mission-critical outdoor use cases and demand more power. More IoT-oriented cellular protocols, such as NB-IoT (Narrowband Internet of Things) and LTE-M (Long-Term Evolution for Machines), are energy-efficient in favourable conditions, however, they still require additional chip investment and mobile infrastructure in licensed spectrum (Labdaoui et al., 2023). In contrast, LoRa modulation aims to achieve long-range with low power consumption with the cost of low bandwidth, which is suitable for sensors and actuators that do not require big payload messaging. LoRa modulation has six Spreading Factors (SF), from SF7 to SF12, which influence data rate, time-on-air, battery life, and receiver sensitivity (Augustin et al., 2016).

[1] https://www.thethingsindustries.com
[2] https://www.chirpstack.io

4.4 Line of Sight

LoRaWAN performance is significantly impacted by line-of-sight (LoS), as the technology relies on low-power, long-range radio frequency communications. With a clear distance between the transmitter and the receiving gateway, LoRaWAN can achieve long range and reliability, often extending several kilometres in rural areas, but can be blocked by obstacles such as buildings and trees, causing attenuation in the communication channel. Urban area deployments require multiple gateways or a routing mechanism to ensure robust network coverage and data transmission due to the constrained LoS (Saban et al., 2021).

Figure 6a has been taken from TTN Mapper[3] and shows the signal heatmap measured in decibels per milliwatt (dBm) of a gateway placed at a 136m-high tower in Hilversum, the Netherlands. Figure 6b shows a simulation of a 902 MHz antenna placed at the same point and altitude using the Radio Mobile Online tool (Roger Coudé VE2DBE, 2024). The correlation between the expected coverage and the effective transmission of the gateway indicates the simulation as a mapping alternative to ad-hoc antenna placement (Lawrence et al., 2018) and how this point of access benefits from the LoS due to the flat topography of the Netherlands.

4.5 MQTT

MQTT is an application layer protocol over TCP (Transmission Control Protocol) specified to be simple to implement, bandwidth-efficient, and provide different levels of Quality of Service (QoS) data delivery (Mishra and Kertesz, 2020). The protocol uses the publish-and-subscribe model, allowing bidirectional communication between millions of clients via an intermediate broker. After a client establishes an acknowledged connection with the MQTT broker, it can publish messages on topics, which are represented by hierarchical keys containing multiple levels separated by slashes. Clients can connect and subscribe to these topics on the broker, which tracks subscribers and forwards messages to them.

MQTT defines three levels of QoS: Level 0 means ”at most one delivery”, which is useful when occasional data loss due to the lack of confirmation is acceptable; Level 1 should be used if the message requires confirmation but allows duplication since it provides ”at least one delivery”; Level 2 has the highest quality level and provides ”exactly once” message delivery, which is recommended for critical applications where data loss and duplication are unacceptable. Higher QoS levels increase resource consumption, such as storage and network traffic, due to the additional steps necessary for confirmation. The protocol also provides features to enhance reliability and security, such as persistent messages and encryption through TLS (Transport Layer Security).

[3]https://ttnmapper.org

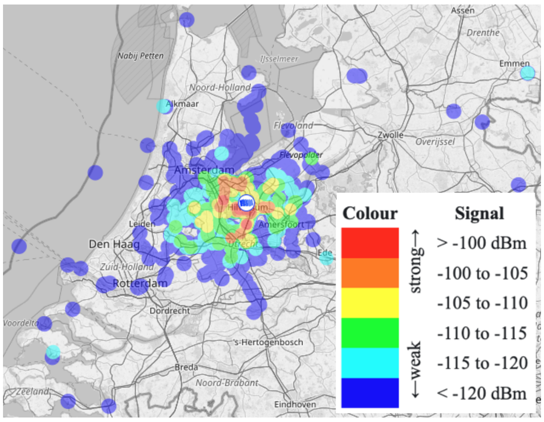
5 DESIGN

In this section, we discuss the design of a multi-network server LoRaWAN artefact, which we called Cold Chain Tracking (CCT), to address the challenges imposed by the cold chain context while tracking perishable goods in remote areas. The stakeholders can check the products’ measured conditions since the end device constantly monitors them during freight. The system also provides essential notifications in case of violations, even if the carrier is not connected to the Internet.

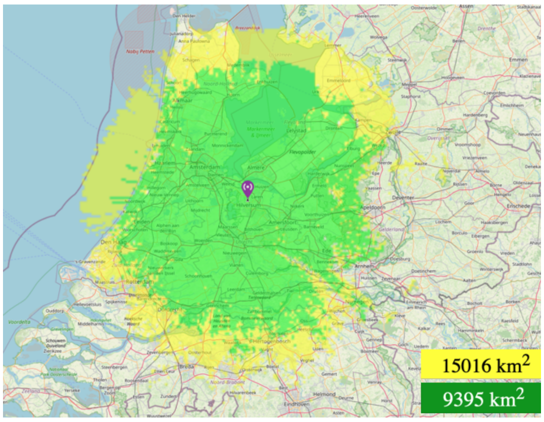
The most valuable novel aspect of this proposal is the support to multiple Network Servers. Our solution expands the coverage area and is beneficial in large-scale deployments in remote zones, ensuring that all devices remain within range of a network server. This potential redundancy helps maintain consistent data transmission and reception, which is critical for applications like cold chain logistics. On top of that, we guarantee the interoperability of the end nodes and how to decouple the application layer from different networks. With these capabilities, this approach offers an alternative for governments and decision-makers to opt for a mixed solution to public-private partnerships (PPP). It enables partnerships with established telecommunications providers to be leveraged and employ private networks in areas where the coverage is not profitable for them.

5.1 Capabilities

Figure 7 illustrates the leading design capability groups of our approach: management, monitoring, alerts, tracking, and security. The management capability encompasses CRUD (Create, Read, Update, and Delete) operations of the main application entities, including measurement types, devices, carriers, products, and freights. Conditions measured by the device sensors, such as temperature and humidity, are decoded, stored, and monitored by the application. CCT identifies and emits alerts for any product condition violation. The tracking capability is based on the LoRaWAN location using the multilateration feature when available. Apart from the encrypted data exchange between the system components, the secu-



(a) Signal heatmap of gateway transmission from TTN Mapper.



(b) 902 MHz antenna coverage simulation.

Figure 6: Heatmap of effective transmissions and coverage simulation for gateway placed at a 136 m tower in Hilversum, NL.

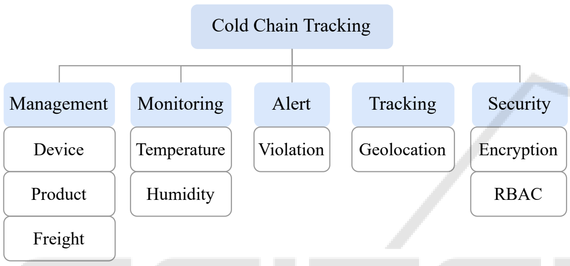


Figure 7: Cold Chain Tracking main capabilities.

rity capability defines the role-based access control (RBAC) for the two main actors (Administrator and Carrier).

Figure 8 shows how the CCT components interact to monitor a freight and alert condition violations. Assuming devices are registered to the Network Server and the CCT application has subscribed to their measurements, the administrator creates a freight by setting up the product details and condition limits, the carrier who will deliver it, the available device to monitor the product conditions, and the origin and destination. The Carriers can list their available freights, choose one of them to start, and turn on the device, which immediately tries to join the server. The device then sends the measurement data to the gateways via LoRa wireless modulation, which forward the received messages to their respective Network Servers. The CCT application then listens to and stores these events, verifying any thresholds violated and sending a downlink in case of a violation. The device then issues an alert to the carrier.

5.2 Architecture

According to the DSM, an architecture structure is a conceptual framework in which each system is an entity that can be decomposed into components that interact to produce overall system behaviour. It supports case-based research, in which we investigate individual cases, study their architecture, identify mechanisms by which overall system-level phenomena are produced, and generalise case by case.

The architecture presented here is similar to the typical LoRaWAN architecture: the CCT system comprises end devices that send uplink messages data using LoRa wireless modulation to gateways connected through backhaul to the Network Server. Figure 9 shows that we adopted a multi-server model instead of a single network server to better align with our design problem. Each provider exposes the device messages to CCT via MQTT topics and provides an HTTPS (Hypertext Transfer Protocol Secure) API (Application Programming Interface) that allows bidirectional communication. Finally, the web application subscribes to the devices' messages via the MQTT broker, storing important information in a PostgreSQL database. The web application uses the Network Server's APIs to send downlink commands to the end nodes when a violation occurs.

We opted for using two network servers that allow the registration and retrieval of devices in the Netherlands, where we carried out this study. KPN Things[4], which is an IoT enterprise solution from the Dutch telecommunications company KPN, was chosen as Network Server due to its high coverage and its device geolocation capability using multilateration (KPN, 2024c). The Things Network (TTN)[5], which is a community network operated by TTS, was chosen to simulate a private deployment since it contains community gateways, thus less coverage, and allows

[4]https://portal.kpnthings.com
[5]https://www.thethingsnetwork.org

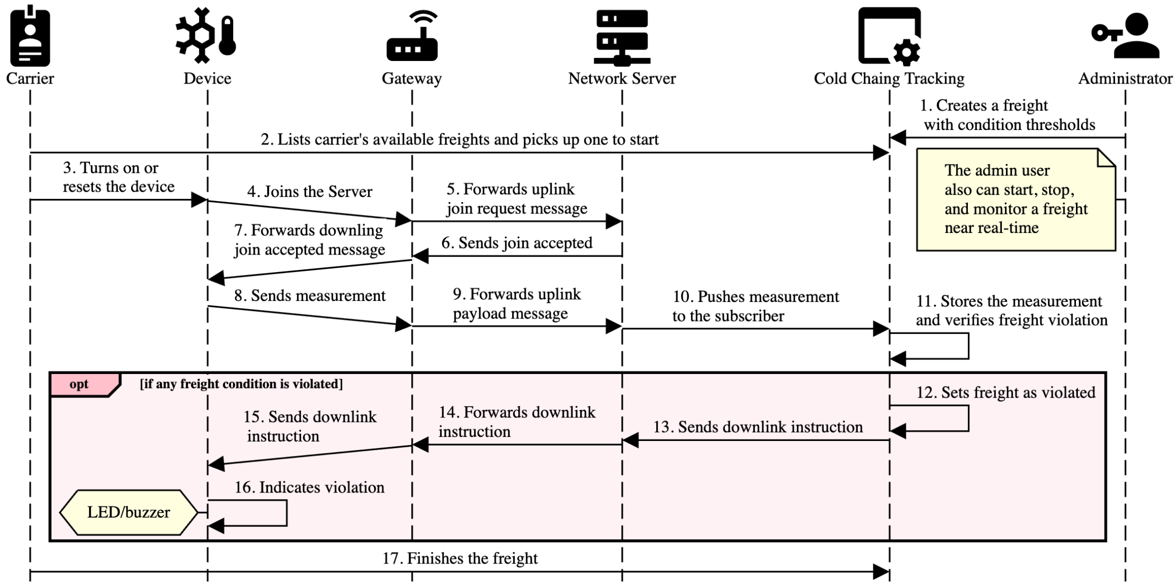


Figure 8: Sequence diagram of the freight feature.

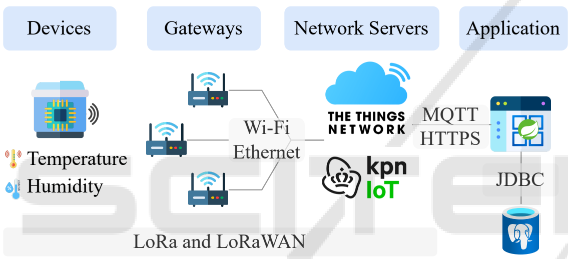


Figure 9: Cold Chain Tracking Architecture diagram.

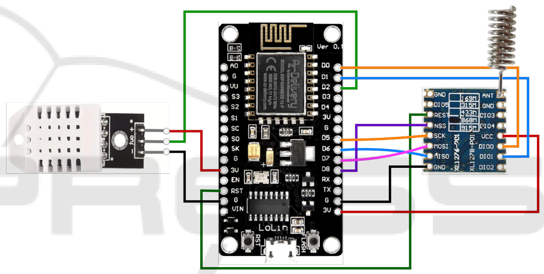


Figure 10: DHT22 and SX1276 modules mapped to NodeMCU.

the addition of custom gateways. TTN applies a Fair Use Policy (FUP), which limits the uplink airtime to 30 seconds per day per node and the downlink messages to 10 messages per day per node.

In DSM, an application of an artefact in a context is called a *treatment*. In the sequel, we describe the technologies involved in the treatment of our problem for each architecture component.

5.3 End Devices

LoRaWAN encompasses various end nodes, including commercial solutions with out-of-the-box sensing capabilities. To allow full customisation and familiarise ourselves with the protocol, we chose to use Az-Delivery NodeMCU v3 (ESP8266) and Heltec WiFi LoRa 32 (ESP32) developer boards with general-purpose input/output (GPIO) pins. We wired up both with a DHT22 module to collect environment temperature (from -40 to +80 ℃) and humidity (0 to 100% Relative Humidity) data. For the former, a LoRa SX1276 module, including an antenna, is wired up along with the sensor, as presented in Figure 10, while the latter already contains it onboarded. Table 1 shows the detailed pin mapping for the NodeMCU v3 board and the wired LoRa and sensor modules. The mapping is more straightforward for the WiFi Lora 32 board since it already contains the LoRa transceiver (only the DHT22 module `OUT (DATA)` pin is wired to the WiFi Lora 32 `D2` pin).

We used Arduino IDE (Integrated Development Environment) to upload the source code[6] to both boards. The source code uses the DHT (Digital Temperature And Humidity) sensor library[7] for reading the sensor digital input, and the LMIC (LoRaWAN-MAC-in-C) library[8], a LoRaWAN Class A and Class B implementation, configured for the European region (863-870 MHz). The board-specific parameters are defined at the beginning of the source code, such as the device keys to join the Network Servers and the pin mapping for the built-in LED (Light-Emitting Diode), LoRa and sensor modules.

[6]https://github.com/alexfabgarcia/cold-chain-tracking
[7]https://github.com/Khuuxuanngoc/DHT-sensor-library
[8]https://github.com/mcci-catena/arduino-lmic

Table 1: NodeMCU wired to SX1276 and DHT22 modules.

NodeMCU v3	SX1276	DHT22
D8 (15)	NSS	
D7 (13)	MOSI	
D6 (12)	MISO	
D5 (14)	SCK	
RST	RST	
D1 (5)	DIO0	
D0 (16)	DIO1	
3.3V	VCC	3.3V (+)
GND	GND	GND (-)
D2 (4)		OUT (DATA)

The `setup` function configures the DHT22 and built-in LED pins, followed by the LMIC initialisation and the first LoRa package sending job, which automatically triggers the join request. The `loop` function only calls the `os_runloop_once` LMIC procedure, which is crucial in the event-driven system that handles LoRaWAN communication. It ensures that operations are handled promptly without blocking the main program flow, such as joining the network, sending data, and receiving acknowledgements.

The constant `TX_INTERVAL` defines the transmission interval, which might become longer due to duty cycle limitations. The `readTemperatureAndHumidty` function reads temperature and humidity through the sensor in the LMIC-encoded format, and then LMIC prepares the upstream data for the next possible transmission time. This invocation occurs whenever the transmission concluded event (`EV_TXCOMPLETE`) is received, and when there is downstream payload, the function `handleConditionViolation` is called to handle the message coming from the Cold Chain Tracking application. The implementation turns on the built-in LED to indicate that a violation has happened.

5.4 Network Servers

KPN Things allows us to provision a set of LoRaWAN nodes for a registered application, including a programmable LoRa device suitable for our purposes, showing the OTAA credentials during registration. The messages that devices send are encoded in the Sensor Measurement List (SenML) format (Jennings et al., 2018) and can be forwarded to multiple destination types, such as custom MQTT brokers or HTTPS endpoints. We used the EMQ free public MQTT broker (EMQ Technologies Inc, 2024) because the network server does not provide a default broker. Finally, KPN Things provides a REST (Representational State Transfer) API that enables external applications to query for devices and send downlink payload to them (KPN, 2024a).

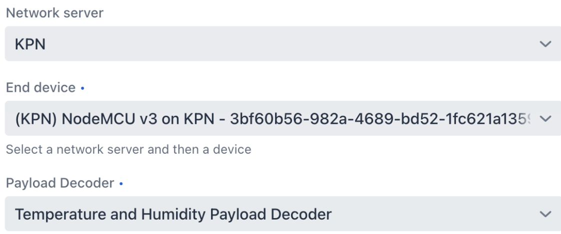


Figure 11: Payload decoder definition for the KPN device.

The TTN configuration is similar to the KPN Things configuration. After setting up the application for the desired region (Europe), we registered the end devices and used the OTAA credentials generated for them. TTN provides an MQTT Broker version 3.1.1 (QoS 0 only) and a REST API (The Things Industries, 2024) for seamless integration. Additionally, TTN allows custom LoRaWAN gateways to be registered. Afterwards, TTN Mapper can be used to show their coverage in a graphic similar to Figure 6a.

5.5 Cold Chain Tracking Application

The CCT web application has been implemented as a Java Spring Boot 3 microservice. Using the Spring Integration framework, it receives data from the MQTT brokers by subscribing message consumers that handle the Network Server's technical details. Vaadin is the platform we used to quickly generate the web interface, including Pro-version components such as Map and CRUD to expedite the development. The Spring Security framework ensures authentication and RBAC authorisation in the web application. A PostgreSQL database stores all the application data, including the device measurements received. Google Maps API (Google Maps Platform, 2024) is used to get the geolocation from the addresses searched.

We leverage the Spring Framework dependency injection mechanism to support multi-network servers with implementations of interfaces such as `NetworkProviderService`. An Administrator user can list the devices from both providers and assign to each of them an available payload decoder, defined by the interface `DevicePayloadDecoder` for extensibility, as displayed in Figure 11. The "Temperature and Humidity Payload Decoder" is the default implementation and decodes the LMIC payload into a key-pair Java `Map` interface implementation.

Figure 12 illustrates the setup of the Novavax COVID-19 Vaccine product on CTT, along with the desirable conditions for its safety. The administrator previously configured the measurement types with their unit (°C or °F). This feature can generalise the design for use in cases other than cold chain since the user can configure multiple measurement types and assign them to product types accordingly.

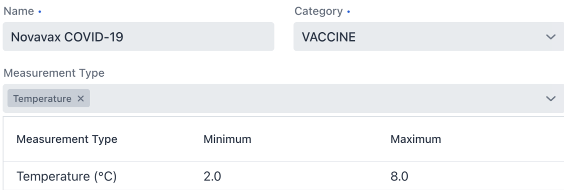


Figure 12: Novavax vaccine restrictions managed on CCT.

After defining all basic entities, the CCT administrator creates a freight by defining the product, the end device, the carrier, a description, the origin and the destination. Figure 13 shows a freight configuration for a Novavax COVID-19 vaccine freight between two vaccination centres.

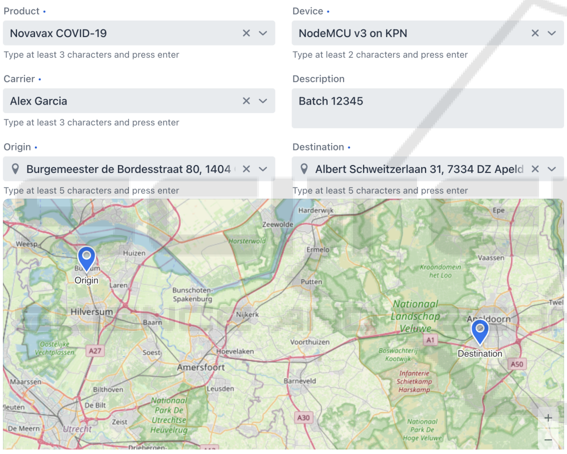


Figure 13: Vaccine freight between two vaccination centres.

The Carrier users can log in to CCT through the end device and start a freight assigned to them. The application MQTT subscribers constantly listen to and handle the device's events with the associated freights. Apart from storing the measurements and location, a downlink message is sent to the end devices via the Network Server's APIs when a violation of the product conditions occurs.

6 TESTS AND VALIDATION

The treatment prototypes were first placed in stationary indoor and outdoor positions to verify the activation of both end devices. Listing 1 shows a sample of the device serial log while handling an `EV_JOINED` containing the network session and application session keys from the OTAA method. Listing 2 shows the log of a device receiving a downlink violation message and turning its LED on.

```
730507: EV_JOINED
devAddr: 15AE(...)
AppSKey: 67-F4-EC-(...)
NwkSKey: AC-AC-6A-(...)
```

Listing 1: OTAA joining serial monitor output sample.

```
819200: EV_TXCOMPLETE (inc. RX waiting)
Condition violated. Turning LED on...
```

Listing 2: Condition violated serial monitor output sample.

We performed JUnit[9] automated tests to assess the integration and capabilities developed in the application layer. With a stable treatment, we carried out a Single-Case Mechanism and a Scale-Up Experiment for the DSM validation phase.

6.1 Single-Case Mechanism

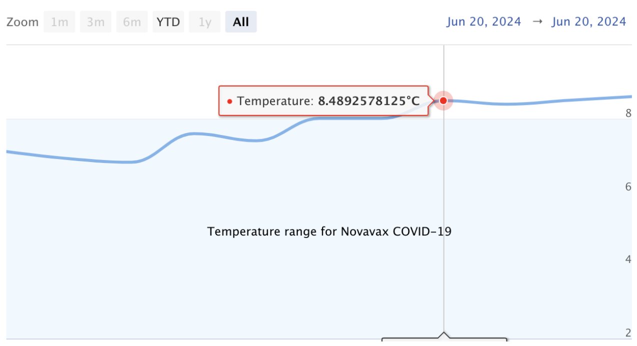
We performed a single-case mechanism validation with several routes and different transport modes to assess the effects caused by the artefact in the context. Figure 14 illustrates the CCT measurements for a freight simulated between Bussum and Soest (two cities in the Netherlands). The temperature metrics shown in Figure 14a match our intentional condition violation, which flagged the freight as violated in CCT and turned the device's LED on. For this same route, Figure 14b shows the approximate device location received from KPN (KPN, 2024c). During our experiments, 90% of the location data received deviate less than 100 meters from the route effectively taken, which is sufficient for applications that do not require high accuracy, so that there is no need to acquire GPS modules.

Figure 15 depicts the transmissions of the NodeMCU v3 connected to the TTN during a sequential route between Soest and Bussum. The blue line indicates the effective route taken, and using the TTN Mapper app connected to the TTN's MQTT broker, we could collect all the metadata and plot the gateways and points for each transmission on the map. The Hilversum Media Park tower gateway received multiple packages, including one from Amersfoort from a distance of 15,6 km, which evidences the impact of the LoS.

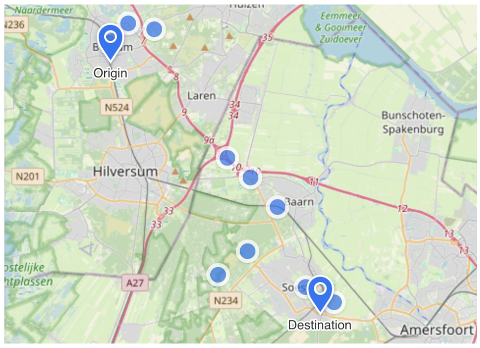
6.2 Scale-Up Experiment

We employed load tests to validate the robustness of our artefact. Being a community network, TTN can-

[9]https://junit.org



(a) Intentional temperature violation.



(b) Approximated location via multilateration.

Figure 14: CCT displaying measurements for freight between Bussum and Soest with WiFi LoRa v3 using KPN.

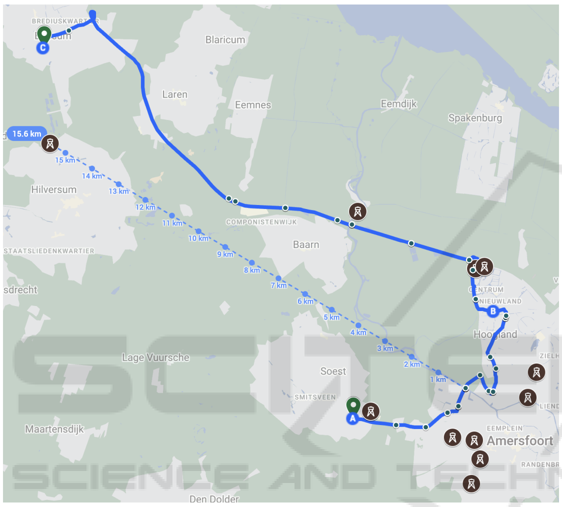


Figure 15: Gateways and longest transmission path mapped using TTN in a route from Soest to Bussum.

not be used for stress tests. At the same time, the KPN Freemium plan only supports up to three devices, even though the Standard enterprise plan supports over one thousand devices (KPN, 2024b). For these reasons, in our performance validation we focused on the LoRaWAN application in response to the stimulus simulated on the MQTT broker.

In an Apache JMeter[10] test plan, we defined 1000 devices for each network server, with 30 minutes of duration and a ramp-up period of 60 seconds, respecting the rate limit imposed by the EMQ public broker. Each device sends approximately one message per minute to the device-specific MQTT topic. The payload sent is a fixed measurement containing the temperature and humidity data, with a 2% chance of being violated temperature conditions, randomly above 8°C.

Besides storing the data received from the end nodes, CCT exposes metrics using the OpenTelemetry[11] standard, which helps validate the CCT performance. We have used Prometheus[12] and Granafa[13], which are powerful open-source tools, to collect and visualise the metrics, respectively.

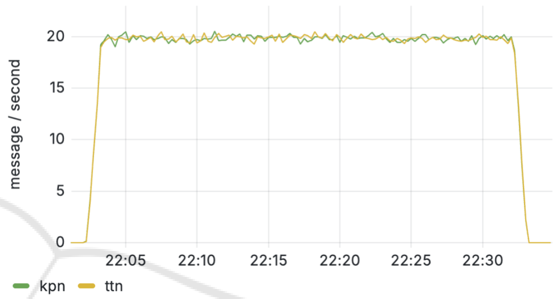


Figure 16: Device measurements throughput per Network Server.

Figure 16 shows a Grafana panel for the throughput of messages handled per second for each network service in the CCT application. The data plotted is compatible with the throughput rate stated in the JMeter test results, namely, 19.8 messages per second. A single application instance achieved this throughput, processing the measurement message in less than ten milliseconds on average.

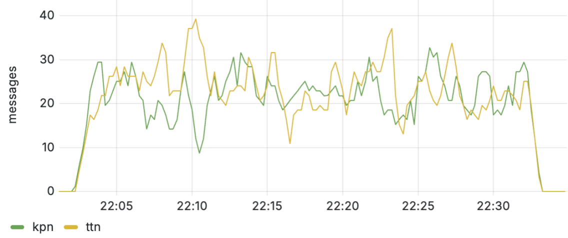


Figure 17: Condition violated metric per Network Server.

Figure 17 displays the violated messages metric exposed by the CCT application per Network Server.

[10]https://jmeter.apache.org

[11]https://opentelemetry.io

[12]https://prometheus.io

[13]https://grafana.com

Considering that the application processed around 2400 messages per minute, it is possible to correlate the error rate proposed in the test plan with the 25 violations identified per minute.

7 FINAL REMARKS

This paper reported on the design, implementation, and in-field tests of Cold Chain Tracking, a multi-network server LoRaWAN application designed to support cold chain logistics. Our study shows that multiple network servers can be used in combination with minimum parameter configuration in the end device due to the LoRaWAN OTAA mechanism and protocol interoperability. In addition, a decoupled and scalable microservice abstracts the source of the device information through multiple MQTT subscribers. This setup offers insights for PPP in various domains where long-range low-power communication excels, allowing new IoT applications to be added by leveraging the same infrastructure and maximising RoI.

The paper also showed how simulation tools can offer a proper balance between antennas' coverage and the LoRa modulation transmissions received by the Network Servers, which can be employed to private gateways deployment as an alternative to ad-hoc attempts. On top of that, the exposed geolocation by multilateration feature is an affordable GPS-free alternative for setups in case a highly accurate location is not required, which means investing more in gateway deployment can pay off compared to high-power end device costs with GPS modules.

Following best object-oriented development practices, the application is suitable and extensible for different network providers, measurement types, and payload decoders to work with the end device's sensing capabilities. This makes this approach in principle appropriate to build systems for other real-world scenarios than cold chain, especially in challenging remote areas.

During the development of our application, we encountered several challenges. Although the LMAC library implements and abstracts most of the particularities of the LoRaWAN protocol for end nodes, finding the correct pin mapping configuration for both boards demanded considerable investigation. Additionally, after wiring the LoRa SX1276 to the NodeMCU v3, not many general-purpose IO pins were left for the sensor itself, and proper mapping also took time. The TTN network does not provide the geolocation by multilateration feature out-of-the-box since it depends on the gateways deployed by the community users. We circumvented this limitation during the test phase by using the TTN Mapper app and a mobile device connected to the Internet. For production deployments, modern LoRa transceivers, proper gateways, and network server configuration must be in place. Further, the TTN MQTT broker version 3.1.1 only supports QoS 0 level, which implies that messages are not guaranteed to be delivered, and the shared subscriptions capability is not supported. This MQTT 5 feature ensures that the broker pushes a message to only one client in a group of subscribers, improving load balancing and fault tolerance while consuming the messages from one topic by the CCT microservice. The TTN webhook integration could be used to forward messages to an MQTT 5 server instead.

We conducted field tests in the Netherlands, where the topography and infrastructure favour the LoRaWAN protocol. Future work could exploit other conditions by applying Technical Action Research (Wieringa, 2014), for example, by a) mapping antennas' coverage using tools such as Roger Coudé's Radio Mobile Online (Roger Coudé VE2DBE, 2024) for remote areas like the Amazon forest for antivenom distribution scenario and b) deploying gateways accordingly in a private network to validate the artefact in more adverse conditions.

Regarding the features implemented in our application, some useful improvements could be to a) automate device registration in the CCT application by employing technologies such as NFC (Near Field Communication); b) send freight thresholds to devices right after they joined the server; c) use the device local storage to handle thresholds when communication is not possible; and d) predict when the conditions will reach thresholds to prevent violations.

ACKNOWLEDGEMENTS

This study was partly financed by the "Coordenação de Aperfeiçoamento de Pessoal de Nível Superior" - Brazil (CAPES) - Finance Code 001. We also thank the Brazilian National Council of Technological and Scientific Development (CNPq) and the São Paulo Research Foundation (FAPESP) for sponsoring our research in the context of the Brazilian National Institute of Science and Technology in Medicine Assisted by Scientific Computing (INCT-MACC).

Regarding software licences, we acknowledge KPN for the KPN Things freemium project, The Things Industries for The Things Stack Sandbox, and Vaadin for the Pro license. We also appreciated the support from the TTN community in the forum.

REFERENCES

Ashton, K. et al. (2009). That 'internet of things' thing. *RFID journal*, 22(7):97–114.

Atzori, L., Iera, A., and Morabito, G. (2010). The internet of things: A survey. *Computer networks*, 54(15):2787–2805.

Augustin, A., Yi, J., Clausen, T., and Townsley, W. M. (2016). A study of LoRa: Long range & low power networks for the Internet of Things. *Sensors*, 16(9):1466.

Aung, M. M. and Chang, Y. S. (2014). Temperature management for the quality assurance of a perishable food supply chain. *Food Control*, 40:198–207.

Badia-Melis, R., Mc Carthy, U., Ruiz-Garcia, L., Garcia-Hierro, J., and Villalba, J. R. (2018). New trends in cold chain monitoring applications - A review. *Food Control*, 86:170–182.

Bose, A., Aithal, A. K., and Rajeshwari, B. (2022). Vaccine cold storage monitoring and tracking using LoRaWAN. In *2022 2nd International Conference on Intelligent Technologies (CONIT)*, pages 1–6. IEEE.

Cattani, M., Boano, C. A., and Römer, K. (2017). An experimental evaluation of the reliability of LoRa long-range low-power wireless communication. *Journal of Sensor and Actuator Networks*, 6(2):7.

Cavalcante, A. M., Marquezini, M. V., Mendes, L., and Moreno, C. S. (2021). 5G for remote areas: Challenges, opportunities and business modeling for Brazil. *IEEE Access*, 9:10829–10843.

Centenaro, M., Vangelista, L., Zanella, A., and Zorzi, M. (2016). Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios. *IEEE Wireless Communications*, 23(5):60–67.

EMQ Technologies Inc (2024). Free public MQTT Broker. https://www.emqx.com/en/mqtt/public-mqtt5-broker. Accessed: 2024-06-20.

Enriko, I., Alemuda, F., and Adrianto, D. (2022). An Approach to Monitor Vaccine Quality During Distribution Using Internet of Things. *Journal of Information Science & Engineering*, 38(5).

Fan, H. W. and Monteiro, W. M. (2018). History and perspectives on how to ensure antivenom accessibility in the most remote areas in Brazil. *Toxicon*, 151:15–23.

Fargas, B. C. and Petersen, M. N. (2017). GPS-free geolocation using LoRa in low-power WANs. In *2017 global internet of things summit (Giots)*, pages 1–6. IEEE.

Frangoudis, P. A., Tsigkanos, C., and Dustdar, S. (2021). Connectivity technology selection and deployment strategies for IoT service provision over LPWAN. *IEEE Internet Computing*, 25(1):61–70.

Garcia, A. F. and de Souza, W. L. (2023). Internet of Things Applications for Cold Chain Vaccine Tracking: A Systematic Literature Review. In *International Conference on Information Technology-New Generations*, pages 323–330. Springer.

Google Maps Platform (2024). Geolocation API Overview. https://developers.google.com/maps/documentation/geolocation/overview. Accessed: 2024-06-20.

Jennings, C., Shelby, Z., Arkko, J., Keranen, A., and Bormann, C. (2018). Sensor measurement lists (SenML). Technical report, Internet Engineering Task Force (IETF).

KPN (2024a). KPN Things Developer Manual - API Access. https://docs.kpnthings.com/dm/concepts/api. Accessed: 2024-06-20.

KPN (2024b). KPN Things Explorer. https://www.kpn.com/zakelijk/internet-of-things/kpn-things/explorer.htm. Accessed: 2024-06-20.

KPN (2024c). KPN Things LoRa Geolocation. https://docs.kpnthings.com/lora/geolocation/intro. Accessed: 2024-06-20.

Labdaoui, N., Nouvel, F., and Dutertre, S. (2023). Energy-efficient IoT Communications: A Comparative Study of Long-Term Evolution for Machines (LTE-M) and Narrowband Internet of Things (NB-IoT) Technologies. In *2023 IEEE Symposium on Computers and Communications (ISCC)*, pages 823–830. IEEE Computer Society.

Lawrence, P. W., Phippard, T. M., Ramachandran, G. S., and Hughes, D. (2018). Developing the IoT to support the health sector: A case study from Kikwit, DR Congo. In *Emerging Technologies for Developing Countries: First International EAI Conference, AFRICATEK 2017, Marrakech, Morocco, March 27-28, 2017 Proceedings 1st*, pages 45–56. Springer.

LoRa Alliance (2024). LoRaWAN Network Coverage - LoRa Alliance. https://lora-alliance.org/lorawan-network-coverage. Accessed: 2024-06-20.

Mishra, B. and Kertesz, A. (2020). The use of MQTT in M2M and IoT systems: A survey. *Ieee Access*, 8:201071–201086.

Roger Coudé VE2DBE (2024). Radio Mobile Online. https://www.ve2dbe.com/rmonline_s.asp. Accessed: 2024-06-20.

Saban, M., Aghzout, O., Medus, L. D., and Rosado, A. (2021). Experimental analysis of IoT networks based on LoRa/LoRaWAN under indoor and outdoor environments: Performance and limitations. *IFAC-PapersOnLine*, 54(4):159–164.

The Things Industries (2024). The Things Stack HTTP (REST) API. https://www.thethingsindustries.com/docs/api/reference/http/routes/. Accessed: 2024-06-20.

Vangelista, L., Zanella, A., and Zorzi, M. (2015). Long-range IoT technologies: The dawn of LoRa™. In *Future Access Enablers for Ubiquitous and Intelligent Infrastructures: First International Conference, FABULOUS 2015, Ohrid, Republic of Macedonia, September 23-25, 2015*, pages 51–58. Springer.

Wieringa, R. J. (2014). *Design science methodology for information systems and software engineering*. Springer.

Zinas, N., Kontogiannis, S., Kokkonis, G., Valsamidis, S., and Kazanidis, I. (2017). Proposed open source architecture for Long Range monitoring. the case study of cattle tracking at Pogoniani. In *Proceedings of the 21st Pan-Hellenic Conference on Informatics*, pages 1–6.