

# Development of a Low-Cost, Power-Efficient Smart Farming Framework using LoRa and IoT

*Atthikumar N<sup>1</sup>, Suvain K K<sup>2</sup>, Mohanraj K<sup>3</sup>, Babu K<sup>4</sup>, Vinoth R<sup>5</sup>, Prabakaran D<sup>6</sup>, and Girmurugan R<sup>7\*</sup>*

<sup>1</sup>Department of Civil Engineering, Nandha College of Technology, Perundurai 638052, Tamil Nadu, India

<sup>2</sup>Department of Agricultural Engineering, Nandha Engineering College, Perundurai 638052, Tamil Nadu, India

<sup>3</sup>Department of Automobile Engineering, Rajalakshmi Engineering College, Chennai 602105, Tamil Nadu, India

<sup>4</sup>Department of Robotics and Automation, Karpaga Vinayaga College of Engineering and Technology, Chengalpet 603308, Tamil Nadu, India

<sup>5</sup>Department of Mechatronics Engineering, Sona College of Technology, Salem 636005, Tamil Nadu, India

<sup>6</sup>Department of Electronics and Communication Engineering, Sri Balaji Chockalingam Engineering College, Arani 632317, Tamil Nadu, India

<sup>7</sup>Department of Mechanical Engineering, Nandha College of Technology, Perundurai 638052, Tamil Nadu, India

**Abstract.** Industrial agriculture is embracing the idea of "smart farming" in response to the rising demands for food throughout the world and the difficulties inherent in traditional agricultural practices. The agri-food supply chain can benefit substantially from smart farming systems' increased efficiency, safety, and productivity because to their high degree of automation and real-time control. This study introduces a tailored smart farming system that utilizes a LoRa and Internet of Things (IoT) based wide-area wireless sensor network that is inexpensive, power-efficient, and has a long range. Existing Programmable Logic Controllers (PLCs) often used in industry and agriculture are connected with LoRa connection in this system. Through Simatic IOT2040, PLCs regulate, control the processes and devices. Agricultural machinery and environmental data can be tracked and managed remotely using this web-based solution. The conversations were automated with users of this messaging service by using the bundled Telegram bot. We have tested the suggested network structure and evaluated the route loss in the wireless LoRa.

## 1 Introduction

"Industry 4.0" refers to the fourth industrial revolution and has recently become a popular topic of study and debate in both academic and professional circles, especially in the fields

---

\*Corresponding Author: [dr.r.girmurugan@gmail.com](mailto:dr.r.girmurugan@gmail.com)

of engineering and management. This study presents an IoT-based Smart Agricultural System that forecasts crop damage by machine learning and ensemble learning techniques [1]. This study systematically examines the ways in which Machine Learning (ML) and Software-Defined Networking (SDN) enhance IoT-based Smart Agricultural Systems [2]. A conceptual approach for digital transformation in Smart Agricultural Systems is provided, highlighting integrated data gathering, sharing, and AI-driven analysis [3]. The study employs biological intelligence optimization algorithms in Smart Agricultural Systems, utilizing IoT and cloud computing for monitoring, remote sensing, and big data analysis. It illustrates enhanced precision by hybrid AI models and LoRa-enabled platforms, suggesting a closed-loop agri-food supply chain that encompasses production, processing and commercialization [4]. This study examines IoT and ML driven Smart Agricultural Systems that provide ongoing monitoring of water, soil, and crop development with minimal human intervention [5]. This research presents a fog-assisted IoT framework for Smart Agricultural Systems, which diminishes latency and enhances efficiency in farm monitoring. The model improves automation, precision and decision-making in smart farming and fortifies the agri-food supply chain by utilizing PLC-based IoT sensing, cloud-fog platforms and LoRa/LoRaWan communication [6]. A cost-effective IoT-driven Smart Agricultural System is presented to optimize soil-water evaporation through rule-based designs. The solution enhances irrigation efficiency, minimizes cloud storage expenses and facilitates scalable precision agriculture for the agri-food supply chain through the integration of PLC controllers, ESP modules, edge-cloud architectures and LoRa-enabled IoT sensors [7]. This study presents UAV-assisted Agriculture 5.0 within Smart Agricultural Systems, augmented by 6G massive MIMO for precision farming. The system employs PLC-IoT integration, cloud computing, and optimized power allocation algorithms to balance throughput and fairness, thereby enhancing UAV-based monitoring and decision-making in the agri-food supply chain [8]. A blockchain-based IoT framework for Smart Agricultural Systems has been created to ensure the security of agricultural data exchange and administration. The integration of PLC automation, cloud computing, and LoRa-based IoT communication, alongside encryption methods and group signatures, improves traceability, identity protection and confidence within the agri-food supply chain [9]. This study introduces an IoT-driven Smart Agricultural System utilizing STM32 controllers and ESP8266 WiFi for precision agriculture. The system, integrated with the OneNET cloud platform, facilitates PLC-driven sensor monitoring, real-time data visualization and remote control, thereby decreasing deployment costs and enhancing effective smart farming within the agri-food supply chain [10].

### **1.1 Contributions and Highlights of the Proposed Model**

Agriculture has a long history of using Programmable Logic Controllers (PLCs) to automate a variety of processes and control machinery; this research work introduces a new way to integrate LoRaWAN connectivity with these older systems. With the suggested solution, smart farms can be easily built by integrating LoRa connection with long-standing automated operations, eliminating the need to replace outdated control systems. A Simatic IOT2040 with a LoRa shield attached to it would establish communication with the current PLC by the Modbus-TCP protocol, allowing for the integration to take place. Through the integration of data collected from several sensor nodes spread out around the farm, we are able to manage the functioning of agricultural machinery such as water pumps.

All of the smart farm system's gadgets may be remotely managed through an intuitive interface due to the cloud server, which processes data and offers a secure, versatile, and scalable web-based platform. The agricultural sector stands to gain a great deal from this novel strategy. The solution can make use of automated processes without replacing costly

outdated control systems by connecting existing PLCs to LoRa networks. Both personnel costs and productivity can be enhanced with the use of remote management capabilities. Additionally, global food production can be improved by the detection and prevention of pests and diseases made possible by real-time monitoring of environmental factors like temperature and humidity.

## 2 Methodology

The "smart farms" are actually new kinds of automated farming systems that make use of the Internet of Things (IoT). The suggested configuration has five primary components: end-node sensors, an Internet of Things LoRaWan gateway, the control hardware, a cloud server housing a web-based platform for management and oversight, and a Telegram bot for a messaging app. This section elaborates on the proposed smart farm system after a brief introduction to LoRa and LoRaWan.

### 2.1 Overview of Lora and LoRawan Technologies

The physical layer of Low-Power Wide Area Networks (LPWANs) is LoRa, while the Media Access Control (MAC) layer is LoRaWAN. Both standards are widely used and accepted worldwide. The LoRa Alliance created LoRaWAN, an open standard. The document delineates hardware, software, and cloud-based elements, illustrating how PLC integration and LoRa/LoRaWan connectivity augment decision-making, boost agricultural precision, and promote sustainable smart farming within the agri-food supply chain as observed in previous studies [11]. Range, duration, and resilience of signals are determined by factors such as transmit power, Spreading Factor (SF), and Code Rate (CR) in the 863-870 MHz ISM band in Europe, where LoRa operates. Using six orthogonal Spreading Factors (OSFs) ranging from seven to twelve, LoRa modulation allows for a trade-off between data throughput, range, and power consumption. When the SF is lower, the chirp rate, data transmission rate, active time of the radio transceivers, and battery life are all improved. Nevertheless, LoRa does have some drawbacks, much like any other technology. The fact that it has restricted bandwidth is a major downside because it means that data transfer speeds could be poor. Additionally, when attempting to send massive volumes of data, LoRa's constant transmission rate can become an issue. Because it uses the same unlicensed spectrum as other wireless technologies, LoRa also has problems with interference. Especially in densely populated areas where radio frequency activity is prevalent, this might disrupt communication. Table 1 shows the advantages and disadvantages of LoRa/LoRaWAN.:

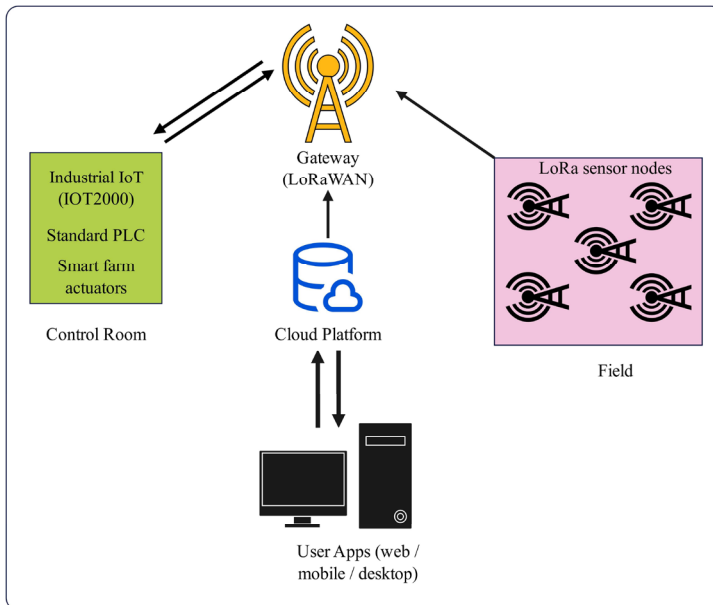
- The LoRa network is protected on two layers by authentication, integrity, and encryption:
- During authentication at the network layer, the end-device and the network server exchange the network session key (NwkSKey), which is a secret key encrypted with AES-128.
- Application layer: The payload communication between end-devices and the application server is protected by an AES-128 secret key called the application session key (AppSKey).
- The LoRa nodes, gateways, and application servers that make up a LoRa-based network's architecture often form a hierarchical topology. Any node in the same LoRaWan network can receive data transmissions from any node in the network. Yet, any gateway in range of a LoRa node the network server in charge of processing the packets will filter out duplicates if they exist.

**Table 1.** Advantages and disadvantages of LoRa/LoRaWAN implementation

Advantages	Disadvantages
Extremely distant Minimal power usage	A slow rate of data transfer Restricted ability
Affordable price	
Safety measures	Unauthorized use of a radio frequency spectrum
The ability to scale and adapt	Integrated design
	Prolonged delay

## 2.2 Design of Smart Farm System Infrastructure

Fig. 1 depicts the two primary networks that make up the suggested smart farm system:



**Fig. 1.** Overview of smart farm system: LoRa gateway linked to cloud server and users

The farm's monitoring network: a collection of LoRa sensors, including moisture and airflow monitors, dispersed wirelessly over the farm to gather the necessary data. The warehouse serves as the control network, where the harvested fruits and vegetables are kept. It is necessary to control the climate of this place, hence it has several environmental sensors, such as a temperature sensor. All of the control equipment, including the air conditioner and irrigation water pump is kept in the warehouse. A PLC is used to control these devices.

It underscores the automation of irrigation with IoT-enabled PLCs and the enhancement of crop monitoring by UAVs, while cloud computing and LoRa-based communication improve productivity, sustainability, and farmer adoption in smart agriculture and the agri-food supply chain [12]. Distributed LoRa sensors across the farm communicate with the central LoRa gateway to report data on environmental factors including humidity, temperature, airflow, and pH.

```

1 // LoRaWan_IOT2000.ino - ABP uplink with CayenneLPP payload
2 #include <WiMODLoRaWAN.h>
3 #if defined(ARDUINO_ARCH_AVR)
4   #include <avr/pgmspace.h>
5 #endif
6 #include <CayenneLPP.h>
7 #include <math.h>
8
9 // ----- LoRa modem serial binding -----
10 #define WIMOD_IF Serial3 // change to Serial/Serial1 as needed
11
12 // ----- LoRaWAN ABP keys -----
13 // TODO: put your values (hex). Example placeholders:
14 const uint32_t DEV_ADR = 0x26011D53; // Device Address (ABP)
15
16 // Network Session Key (16 bytes)
17 const unsigned char NWKSKY[16] = {
18     0x89,0x2C,0x83,0x07,0x5C,0x03,0x8F,0x2A,0xBF,0x2A,0xAC,0x9E,0x8B,0xB6,0x3F,0xB5
19 };
20 // App Session Key (16 bytes)
21 const unsigned char APPSKY[16] = {
22     0x98,0x20,0x96,0xC3,0x7C,0xA3,0x61,0x0B,0x3B,0xC8,0xB0,0xC6,0xA1,0x3C,0x7D,0x2E
23 };
24
25 // ----- Types/State -----
26 typedef enum TModemState {
27     ModemState_Disconnected = 0,
28     ModemState_ConnectRequestSent,
29     ModemState_Connected,
30     ModemState_FailedToConnect
31 } TModemState;
32
33 typedef struct TRuntimeInfo {
34     TModemState ModemState;
35 } TRuntimeInfo;
36
37 WiMODLoRaWAN wimod(WIMOD_IF); // modem handle
38 TRuntimeInfo RIB = { ModemState_Disconnected };
39
40 static uint32_t loopCnt = 0;
41 static TWIMODLORAWAN_TX_Data txData;
42
43 // Cayenne LPP buffer and objects
44 CayenneLPP lpp(64);
45
46 // ----- Sensor stub (replace with your real reads) -----
47 float readTemperatureC() {
48     // e.g., DHT22/BME280; here a dummy sawtooth
49     return 24.0 + (millis() % 10000) / 1000.0; // 24..34 °C
50 }
    
```

**Fig. 2.** Code snippet for LoRaWAN on SIMATIC IoT2040

The internet-connected gateway transfers data collected by the sensors to a server in the cloud, where it is processed and eventually saved in a database. Actions can be configured manually through the web platform or automatically executed based on sensor data; the gateway also delivers server control commands to the warehouse PLC. For example, if the soil needs more water, the PLC will activate the water pump. By specifying the desired device and the values at which the actuator is activated, the user can setup and set the automatic commands through the web platform. Sufficiently dispersed LoRa end-nodes and a high-altitude location for the LoRa gateway will provide optimal communication coverage. A gateway connects Internet of Things devices to a server or cloud.

The gateway is in charge of gathering information from various devices and sensors, analyzing it, and then transmitting it to a server or cloud for further processing. Some of the protocols used by gateways in the Internet of Things are MQTT, CoAP, HTTP, and DDS. The LoRaWAN protocol is utilized by the LoRa gateway in LoRaWan technology. It is accountable for controlling the flow of data and security between the end devices and the LoRa gateway. Because of its low power consumption and ability to support long-distance

communication, it is perfect for Internet of Things (IoT) applications. Our system's LoRa connectivity was set up using the "WiMOD LoRa Lite Gateway" from IMST. The control unit for the farm's machinery, including the water pump, is kept in the storage facility. This PLC needs to be LoRa-connected wirelessly so it can communicate with the gateway and get instructions from the server in the cloud. The IOT2040 was linked to a standard S7 PLC over Modbus TCP in order to accomplish this. Automatic execution based on sensor inputs or manual configuration through the online platform are both made possible by the gateway, which sends control commands from the server to the warehouse PLC. Current PLCs can be linked to the LoRa network with this approach. The WiMOD LoRAWAN library, which is essential for linking a WiMOD module to an Arduino board, is pre-installed on the Shield. This review explores IoT-based energy-efficient secure routing methods for Smart Agricultural Systems. By integrating PLCs, cloud computing, aerial imaging, and LoRa/LoRAWAN networks, it highlights improved resource optimization, monitoring, and supply chain security for sustainable agriculture[13]. The shield may interface with the main board of the IOT2040 through its two UART ports. Fig. 2 explains how to add the Intel Galileo firmware to the Arduino IDE so that to program the LoRa communication shield. To enable reading of LoRa payloads transmitted by the LoRa shield board between the IOT2040 and the gateway, we have included the "node-red-node-Arduino" package in the NodeRed dashboard.

In the warehouse, the following components are controlled by the PLC:

- Irrigation water volume is managed by a water pump and its corresponding valves.
- Refrigerators, freezers, and air conditioners all work together to keep perishable goods at just the right temperature and humidity for as long as possible.
- Protection against intrusion.
- Program the lights to turn on and off in order to simulate presence and nighttime activities.
- Temperature sensor with LoRa protocol.
- Wireless humidity sensor with LoRa protocol.

The LoRa gateway and the PLC are both managed by the IOT2040 over Modbus-TCP. The remaining farm sensors connect to the LoRa gateway in a straightforward fashion as LoRa end-nodes. The IOT2040 and the web app were both enhanced with the help of a bot that communicated over the Telegram Bot API. An account was created with @botfather in order to create the Telegram bot. To begin using @botfather, username, command and bot name was created using the command /newbot. After the bot was built, a bot token was utilized for Telegram communication with our system. The initial step in integrating the Telegram bot with the IOT2040 was to install the "node-red-contrib-telegrambot" package from NodeRed. Direct user control commands, like turning on the lights, are the integration of all about it. A JSON code was used to define many instructions and their behaviors on the Telegram node. The user can thus issue a command using Telegram, and the IOT2040 will process it, confirm with the user, and then send a message thereafter. In order to avoid the duty cycle lag that comes with sending the most current reading to the database, this second integration handles user requests that pertain to the database or sensor nodes.

For example, it can process queries for the most up-to-date irrigation water consumption or the timing of the most recent event where lights were turned off. In this study, LoRAWAN communication is implemented using the Things Network (TTN). Through its open-source nature, TTN facilitates the free exchange of data between LoRa end-nodes and gateways that adhere to the LoRAWAN specifications. In an effort to collectively expand the network, several people from all over the globe have recently registered thousands of gateways in the TTN platform. Users can link their Internet of Things LoRa devices to the TTN network's

preexisting gateways. For those who like to use their own unique platform, connecting to a private server is also an option.

### 2.3 Experimental Test

An end-node that integrates an, temperature sensor, altimeter, accelerometer and GPS module the LoRa Mote II from IMST was utilized in this test. Approximately 24 meters above ground, on the sixth floor of an apartment building, stood the LoRa gateway. Our LoRa end-node device was set up as follows:

- Class A LoRa Network Class
- Timing: 867.1 to 868.5 MHz
- A 125 kHz bandwidth
- A spread factor (SF) of seven
- Ratio of Coding: 4/5
- Power at Transmission: +14 dB
- Up 2 dB for the transmitting antenna
- Gain at the receiver antenna: +2 dB

Ten messages were sent at each site. The maximum distance was achieved by fixing the Spreading Factor (SF) to 7. Instead of fixing the SF, it is better to use the ADR mechanism in similar applications. This will optimize airtime, power consumption and data rates which will increase network's capacity and range.

By plugging the values of NACK (number of correctly recognized packets) and NAP (total number of packets transmitted) into Equation (1), we can determine the Received Packets Percentage (RPP) for every site.

$$RPP[\%] = 100 \times \frac{NACK}{NAP} \tag{1}$$

Learning about path loss (PL) is crucial for comprehending how a LoRa signal's power degrades as it passes across different environments. Both the prediction of a LoRa network's range and the design of efficient communication systems can benefit from this data. Learning about the elements that affect LoRa route loss—like signal frequency, distance from transmitter to receiver and kind of environment—will help. A better distribution of end-nodes, the prediction of a signal's power loss as it propagates through a medium, the identification of possible problems with a LoRa network and suggestions for improving its performance can all be achieved by PL modeling. In natural settings, large-scale fading (PL) assessed by signal-to-noise ratio (SNR) and RSSI, where  $P_t$  is the transmission power,  $G_t$  is the transmitter gain, and  $G_r$  is the receiver gain

$$PL = P_t + G_t + G_r + 10 \times \log_{10} \left( 1 + \frac{1}{SNR} \right) - RSSI \tag{2}$$

In a free-space setting, the Fris law (Equation (3)) shows the received power,  $P_r$ . to transmitted power,  $P_t$ . Here,  $G_t$  and  $G_r$  denote the transmitter's and receiver's gains,  $\lambda$  the wave length, and  $d$  the distance between the two.

$$\frac{P_r}{P_t} = G_t G_r \left( \frac{\lambda}{4\pi d} \right)^2 \tag{3}$$

A reference distance and a path loss exponent  $\gamma$  are added to a non-free space environment. Next, we get Equation (4) from Equation (3).

$$\frac{P_r}{P_t} = G_t G_r \left( \frac{\lambda}{4\pi} \right)^2 \times \frac{1}{d_0^\gamma} \times \left( \frac{d_0}{d} \right)^\gamma \tag{4}$$

One way to estimate PL is to model the experimental PL using the first-order fit. Then, using Equation (5), we can get the Estimated Path Loss (EPL).

$$EPL = PL_0 + 10 \times \gamma \times \log_{10} \left( \frac{d}{d_0} \right) \tag{5}$$

Here,  $d_0 = 1$  m, and  $P_0$  stands for the PL intercept at reference distance  $d_0$  (1-100 m) in microcellular systems. The propagation environment's measurement findings can be used to determine the path loss exponent  $\gamma$ . A model fitting the measured data can be used to extract the required parameters, such as the route loss exponent, allowing for this to be accomplished. Different types of propagation environments and different types of information may necessitate different methods. Gamma is usually 2 in free space. Depending on the building's structure, materials, and barriers, it can range from 1.6 to 6, although in urban surroundings it ranges from 2.7 to 3.5. As demonstrated in Equation (6), shadow fading is the root cause of discrepancies between real PL and estimated PL (EPL). Several statistical indicators including as range, standard deviation and mean, are used to exhibit PL data in order to give a more thorough evaluation. Table 2 summarizes the EPL parameters and their accompanying statistical indicators.

$$\sigma_{SF} = std(PL - EPL) \tag{6}$$

**Table 2.** Statistical analysis of lognormal PL model parameters outdoors

EPL exponent ( $\gamma$ )	2.4
EPL intercept (PL( $d_0$ ))	35.14 dB
Standard deviation ( $\sigma_{SF}$ )	4
Standard error of the mean (SEM)	1.11
Range	71.09 dB

### 3 Results and Discussions

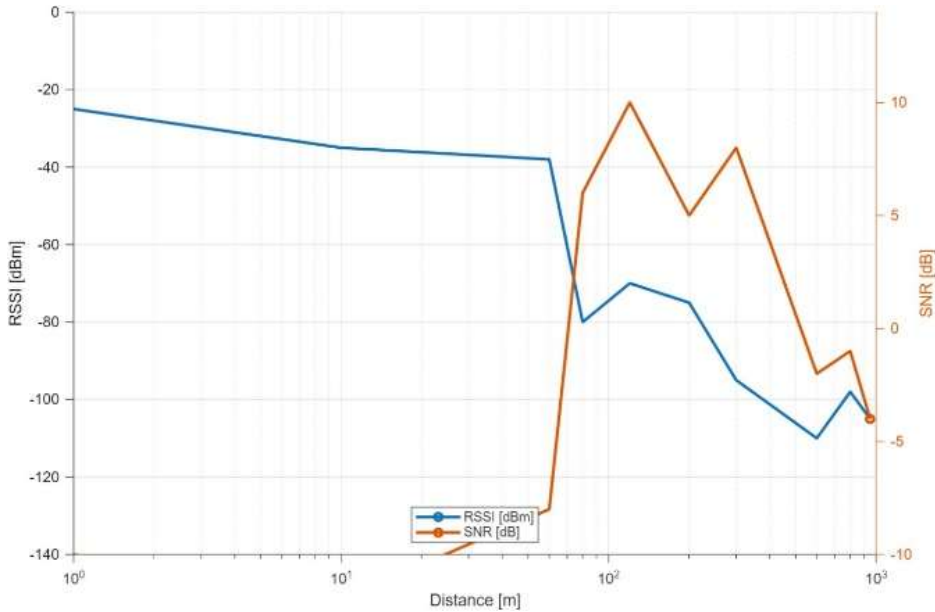
In the previous test, measurement points with a Received Packets Percentage (RPP) below 50% were omitted.

**Table 3.** Received packet metrics (SNR, RSSI, RPP) across measurement points

Location	SNR (dB)	RSSI (dBm)	Distance (m)	RPP (%)
P1	10	-34	55	100
P2	6.0	-82	125	98
P3	7.8	-72	210	100
P4	8.5	-78	295	99
P5	5.5	-92	420	98
P6	6.5	-102	560	88
P7	4.8	-97	590	82
P8	3.0	-108	650	55
P9	-1.5	-107	725	62
P10	-2.5	-99	760	72
P11	-4.2	-111	800	63

For each valid point, Table 3 presents the signal-to-noise ratio (SNR), Received Signal Strength Indicator (RSSI) and RPP values. Fig. 3 shows how the RSSI and SNR change with distance. It is clear that the RSSI drops off with increasing distance. The initial packet loss occurred at position P6 (100 m); the RSSI dropped below -80 dBm starting from position P5 and onwards. In regards to SNR, it exhibited a comparable pattern: the SNR value remained greater than 0 dB until P9 (200 m), it is dropped to -3 dB at P11 (85 m). At P8 (95 m), the rate was the lowest, while at P11 (795 m), 60% of the packets were received. Based on the results of this test, end-nodes and the gateway can provide successful communication at

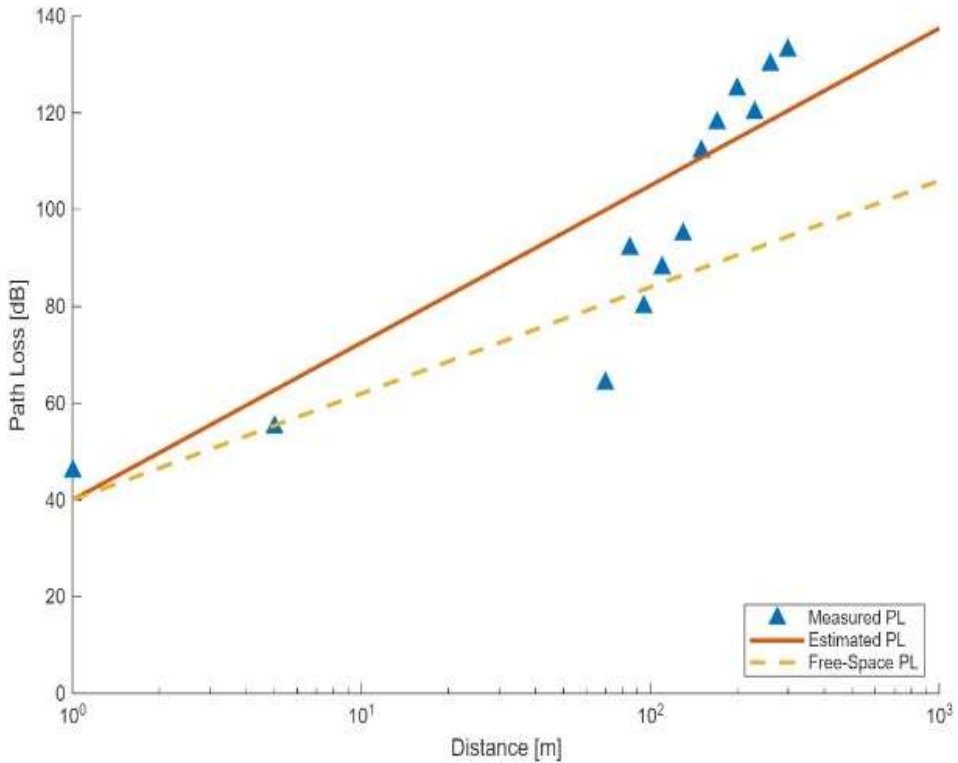
distances less than 500 m. The PL was determined by applying Equation (2) to the experimentally obtained data in Table 3. Comparisons with the Free-Space Path Loss (FSPL) and Estimated Path Loss (EPL) models are shown in Fig. 4. As the distance increases, these results suggest that the proposed approach for estimating the route loss becomes increasingly valid. When looking at distances less than 80 meters, the suggested model reveals instability, primarily due to shadowing effects. It is possible to receive packets with a PL below 125-130 dB. The lowest attenuation is observed in the free space model FSPL, which is not surprising.



**Fig. 3.** Outdoor LoRa test: Avg. RSSI and SNR vs. distance

This means that, there are usually only certain settings, frequency bands, and antenna setups where the PL model parameters will work. Therefore, the suggested PL estimate model can be used to enhance the energy efficiency and reliability of LoRa mesh networks in agricultural settings, under conditions comparable to our practical test. One of the reasons LoRa can achieve long-range communication is that it is designed to operate at very low power levels. On the other hand, this makes the signal more vulnerable to environmental interference and noise, which can lead to an abrupt and substantial rise in PL. Many environmental conditions, such as buildings, trees, and other wireless devices, might cause the LoRa signal's route loss to suddenly increase. A substantial rise in PL can occur as a result of signal absorption, scattering, or directional reflection caused by these causes. When developing smart farming solutions, it is crucial to take LoRa's transmission range into account.

Several factors affect the range of LoRa, including the frequency, transmission power, terrain, and barriers. The typical LoRa transmission range is several hundred meters in densely populated areas and several kilometers in more rural ones.



**Fig. 4.** Comparison of estimated and measured PL with FS model in farm setting

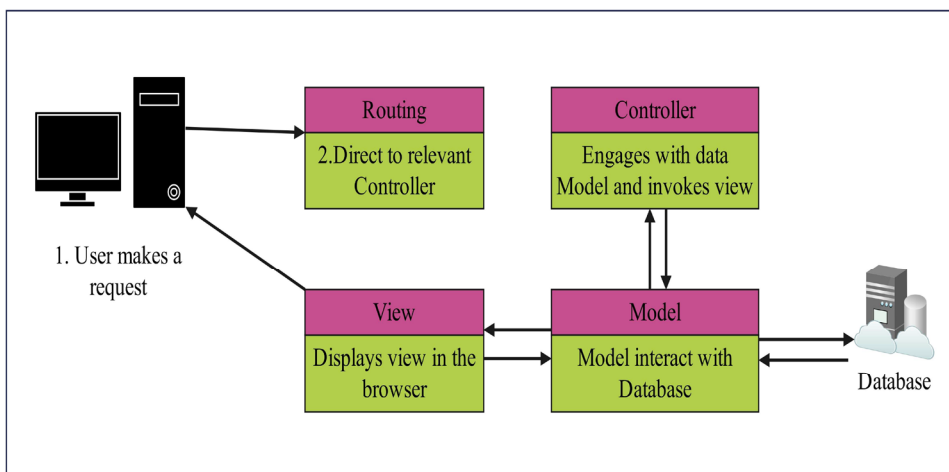
The vast majority of smart farming applications fall inside this spectrum. Nevertheless, there are situations where expanding LoRa's broadcast range is necessary to cover more ground. A network of LoRa gateways, which communicate with both the end devices and the command center, can do this. LoRa gateways can receive and transmit messages over great distances, allowing end devices to have a longer transmission range. Recent studies have demonstrated the Integrating plant physiology with cloud computing, AI-driven analytics, and LoRa-based IoT platforms, it offers a comprehensive strategy for sustainable smart farming and precision irrigation within the agri-food supply chain [14]. Finally, LoRa shows great promise as a technology for smart agricultural applications. An excellent choice for precision agriculture, its long-range, low-power transmission allows for the widespread gathering of data from a variety of sensors and devices.

## 4 Design of the Web Monitoring Platform

A customized application is necessary for the described smart farm system in order to allow for more flexible management, data customization, and analysis. Since relational databases make it easier to manage and maintain the integrity and quality of data over time, we opted to use one.

This is because characteristics like scalability, data consistency, query flexibility, and support for transactions all contribute to the application's long-term viability and reliability. A non-relational database is another option that could be considered for this task. The Bootstrap framework is used to build the front-end of the web app. It has a responsive design that works

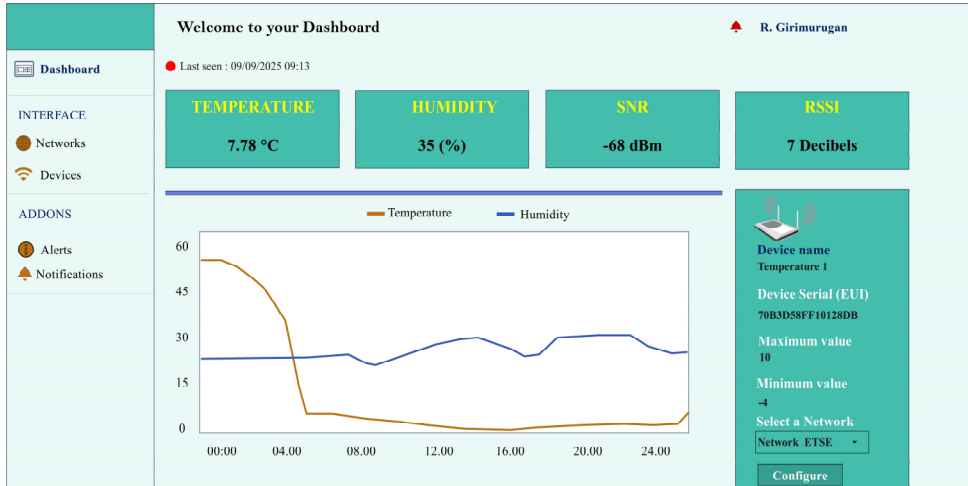
with different screen sizes. In addition to remotely controlling the machines linked to the PLC, the web application makes it easy to retrieve and evaluate data gathered by the sensor nodes. In addition, it comes with an intuitive UI that lets set up the system preferences and create your own notifications and alerts. Anyone with an internet connection can access the web application because it is housed on a cloud server. In previous studies employs biological intelligence optimization algorithms in Smart Agricultural Systems, utilizing IoT and web based monitoring, remote sensing, and big data analysis [15]. The presentation layer and the application logic are separated in Laravel by using the Model-View-Controller (MVC) architecture. Methods used by the Model-View-Controller design. Fig. 5 depicts the routing procedure, which entails receiving a request and sending it to the correct controller. With route model binding can easily insert model instances into routes automatically. A controller's job is to process user requests and use models to access or save data in the database; views then use this data. By default, Laravel resolves Eloquent models expressed in controller actions or routes and passes their values as parameters to controller methods using its implicit route model binding feature. In this project, a MySQL database is used to manage data across various tables. The Bootstrap framework and AngularJS, which offer CSS classes and JavaScript functions, work together to create a responsive user interface that works with a wide range of screen sizes. A user-friendly navigation mechanism and graphical information, such as tables and charts, are integrated into the interface to provide historical data. The charts were made using the free and open-source toolkit. Users can also export data in various forms from the platform, which can be used by offline software tools for additional research.



**Fig. 5.** Web monitoring platform flow from user request to system response

The receiving payloads are forwarded directly to our cloud server and stored in the database through communication between the LoRa gateway and the TTN server. This allows for visualization and monitoring through the user interface. Just like any other data, user orders must first go by the TTN server before reaching the LoRa gateway, which in turn contacts the LoRa end-node. Each user can register several networks that correspond to their related farms using the web application. There are actually a lot of interconnected devices in each network, or farm, and users manage all of them from a single interface. Since all data passes through the TTN LoRaWan network before reaching our cloud server, users must first register their devices in the TTN server before they can use their identification to register them in the web application's dashboard page. The main page and control panel of the web app are shown in Fig. 6. The maximum and minimum values was configured for particular devices and other

settings for each LoRa end-node on the dashboard. Tables and figures display the data acquired by the sensor nodes, and users have the option to export the data as an Excel or CSV file for further analysis, both basic and advanced. The web app's MVC architecture, which is based on the Laravel framework also makes it easy to add new features. The web application provides a thorough and easy-to-use interface for monitoring and evaluating the data collected by the smart farm system.

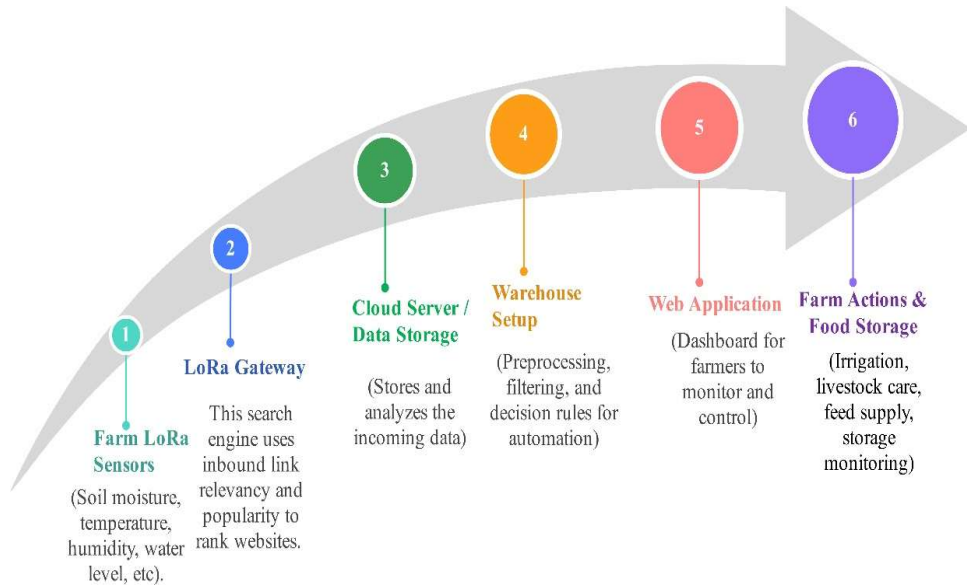


**Fig. 6.** Dashboard view of Mi Granja with temperature sensor data

This web app was found on a server in the cloud, and its key characteristics are:

- Design that can be scaled, lightweight, and responsive quickly.
- Robust safety features made possible by the MVC architecture and authentication mechanism.
- The web application is accessible from a variety of devices, providing users with a multi-device experience.
- Users are notified instantly by email and Telegram.
- Numerous graphs and charts for data visualization.
- Searchable data stored in an organized manner.
- Create CSV and Excel files to export data for more in-depth analysis.

Fig. 7 provides an overview of the system that is being explained. Distributed LoRa end-node sensors gather data about the farm's environment as part of the smart agriculture system. A LoRaWAN gateway receives data from sensors and then transmits it to a server in the cloud. A PLC, or Programmable Logic Controller, is also a part of the system; it controls the lighting and water pumps in the warehouse, among other machinery. Through Modbus-TCP, the IoT2040 is linked to the PLC, and Node-RED is used for programming. A web-based monitoring tool gives users access to the system, letting them remotely manage the warehouse machines through the IoT2040. A Telegram bot also allows users to make requests and issue orders. After receiving data, the cloud server processes it and saves the results.



**Fig. 7.** Smart farming IoT system with PLCs and LoRa nodes linked through a gateway to a cloud-hosted web application

## 5 Conclusions

An all-inclusive Internet of Things (IoT) solution for a smart farm is described in this study. By keeping an eye on and controlling humidity and temperature, the technology hopes to make food product storage better. Automating processes like watering and temperature regulation is also possible with its remote monitoring and control capabilities by a web-based application. The goal of the created web app is to make building and administering IoT networks easier. Access to data analysis and download capabilities, as well as the opportunity to add devices, are provided to users. Using the MVC design pattern guarantees the application's scalability and ability to add new features as required. It also makes it easier to tweak and update. The app's responsive design helps it work flawlessly on any device, regardless of screen size. The software is perfect for administering numerous sensor networks because to its minimal server resource usage, which enables it to accommodate a large number of concurrent users. The system expands its functionality by incorporating new components that improve upon the conventional LoRa IoT network architecture. These components include a LoRa interface that allows data exchange with farm-based automation PLCs and an integration with Telegram that allows direct user communication through a popular messaging platform.

## References

1. N. G. Rezk, A.-F. Attia, M. A. El-Rashidy, A. El-Sayed, E. E.-D. Hemdan, An efficient IoT-based crop damage prediction framework in smart agricultural systems. *Computers and Electronics in Agriculture*. **15**, 27742 (2025). <https://doi.org/10.1016/j.compag.2025.109876>

2. S. Amrani, K. Medani, C. Gherbi, H. Mabed, Machine learning application in software-defined networking-based agricultural Internet of Things: a systematic literature review. *Cluster Computing*. **28**, 647 (2025). <https://doi.org/10.1007/s10586-025-04035-1>
3. G. Gebresenbet, T. Bosona, D. Patterson, H. Persson, B. Fischer, N. Mandaluniz, G. Chirici, A. Zacepins, V Komasilovs, T Pitulac, A. Nasirahmadi, A concept for application of integrated digital technologies to enhance future smart agricultural systems. *Agricultural Systems*. **5**, 100255 (2023). <https://doi.org/10.1016/j.agry.2023.103184>
4. Z. Chen, Z. Liao, D. Qian, J. Li, Design and analysis of intelligent agricultural monitoring system based on biological intelligence optimization algorithm. *Scientific Reports*. **2022**, 18759 (2022). <https://doi.org/10.1038/s41598-022-23456-7>
5. A. A. S. Ahamed Ali, J. Justindhas, M. Alagarsamy, K. Karthikram, J. Sakkarai, K. Suriyan, A comprehensive analysis on IoT-based smart farming solutions using machine learning algorithms. *International Journal of Advanced Computer Science and Applications*. **11**, 1550 (2022). <https://doi.org/10.14569/IJACSA.2022.0110317>
6. G. Singh, J. Singh, S. Juneja, Y. Gulzar, D. Gupta, K.Z. Ghafoor, M. Kumar, Empowering self-sustained agriculture: an IoT field tracking model driven by fog computing. *Journal of Ambient Intelligence and Humanized Computing*. **5**, 1 (2025). <https://doi.org/10.1007/s12652-025-04110-5>
7. D. Gupta, A. K. Dasgupta, K. Chakraborty, A. Chakrabarti, Cost-efficient rule-based soil-water kinetics driven automated architecture for smart agriculture. *IEEE Internet of Things Journal*. **6**, 660 (2025). <https://doi.org/10.1109/IJOT.2025.3158907>
8. A. Massaoudi, W. Aydi, Optimized power allocation algorithms for UAV-based agricultural systems. *Computers and Electrical Engineering*. **13**, 103292 (2025). <https://doi.org/10.1016/j.compeleceng.2025.103292>
9. M. Xie, J. Liu, S. Chen, M. Lin, G. Xu, Z. Xu, Z. Chang, Traceability and identity protection in smart agricultural IoT system framework based on blockchains. *IEEE Access*. **22**, 5335 (2025). <https://doi.org/10.1109/ACCESS.2025.3341187>
10. L. Guo, I. M. Mohd Yassin, Z. I. Ismael Rizman, Application of information collection and control system based on the Internet of Things in smart agriculture. *Sensors*. **11**, 241 (2024). <https://doi.org/10.3390/s2411241>
11. D. A. Gzar, A. M. Mahmood, M. K. A. Al-Adilee, Recent trends of smart agricultural systems based on Internet of Things technology: a survey. *Computers and Electronics in Agriculture*. **104**, 108453 (2022). <https://doi.org/10.1016/j.compag.2022.107630>
12. M. A. B. Rodzoan, A. Shah, Smart farming: integrating IoT and UAV technologies for precision agriculture through the lens of technology acceptance and the UTAUT2 model. *Sensors*. **22**, 6218 (2024). <https://doi.org/10.3390/s22216218>
13. C. S. Kumar, R. V. Vijay Anand, A review of energy-efficient secured routing algorithm for IoT-enabled smart agricultural systems. *Wireless Networks*. **48**, 339 (2023). <https://doi.org/10.1007/s11276-023-03895-2>
14. E. Amirinezhadfar, A. Niazi Tabar, M. Bashir, W.-C. Yang, Plant osmosensors in next-generation smart agriculture: from innovation to application. *Trends in Plant Science*. **234**, 121607 (2025). <https://doi.org/10.1016/j.tplants.2025.234>
15. S. O. Araújo, R. S. Silva Peres, L. B. Pian, F. Lidon, J. C. Ramalho, J. Oliveira, Smart agricultural system using proximal sensing, artificial intelligence and LoRa technology: a case study in vineyard management. *Computers and Electronics in Agriculture*. (2024). <https://doi.org/10.1016/j.compag.2024.109978>