electronics

MDPI

Article

Test Platform for Developing Processes of Autonomous Identification in RFID Systems with Proximity-Range Read/Write Devices

Bartłomiej Wilczkiewicz , Piotr Jankowski-Mihułowicz * and Mariusz Węglarski *

Department of Electronic and Telecommunications Systems, Rzeszów University of Technology, 35-959 Rzeszów, Poland

* Correspondence: pjanko@prz.edu.pl (P.J.-M.); wmar@prz.edu.pl (M.W.); Tel.: +48-1785-44708 (P.J.-M. & M.W.)

Abstract: The subject of a distributed RFID system with proximity-range read/write devices (RWD) is considered in this paper. Possible work scenarios were presented in the scope of industrial implementations and were then tested in a dedicated laboratory set. The development system is based on a high-frequency RWD integrated with a Wi-Fi microcontroller unit to create an Internet of things connected with a server (for data exchanging, user interface, etc.) via a wireless local area network. In practical applications, in order to increase the interrogation zone (IZ), there is a tendency to use one RWD with significant output power equipped with a multiplexer for managing several antennas located in the operational space. Such a solution is often economically unprofitable and even impossible to implement, especially in the case of the need to create the large IZ. Responding to market demand, the authors propose a distributed system developed on the basis of several cheap RFID reader modules and a few freely available hardware/software tools. They created the fully functional RFID platform and confirmed its usefulness in static and dynamic systems of object identification.

Keywords: RFID; MQTT; node-red; distributed RFID system; internet of things; wireless network

Citation: Wilczkiewicz, B.; Jankowski-Mihułowicz, P.; Węglarski, M. Test Platform for Developing Processes of Autonomous Identification in RFID Systems with Proximity-Range Read/Write Devices. *Electronics* **2023**, *12*, 617. https://doi.org/10.3390/electronics12030617

Academic Editors: Mohammed Abdulhakim Al-Absi and Rui Fu

Received: 30 December 2022
Revised: 19 January 2023
Accepted: 24 January 2023
Published: 26 January 2023

1. Introduction

The designers of automatic object identification systems have at their disposal a great selection of various RFID (radiofrequency identification) devices so that they may create new solutions matching the progress and growing requirements of modern industry, broadly understood logistics, the distribution of goods and trade, access control, diagnostics, payments, etc. Unfortunately, when selecting the system equipment, in most cases, they have to primarily take into account the economic aspects of the target application. Otherwise, the developed system would meet the technical requirements but would not be accepted by the client. Regardless of the operating band selected for the application (LF: 125 kHz, HF: 13.56 MHz, UHF: 860–960 MHz), it must always be ensured that the operational space is maximally covered by the interrogation zone (IZ) [1].

The IZ is the basic determinant of the efficiency of the designed RFID system. In order to maximize the IZ, read/write devices (RWD) with high output power, equipped with antenna multiplexers, are most often selected. By locating antennas at various points in the operational space, it is possible to create even more extensive and complex zones of object interrogation. Manufacturers of RFID devices offer many different constructions, the price of which depends primarily on the nominal parameters, i.e., operating frequency band, the range of output power, the sensitivity of the RF front-end, as well as the availability of software tools or application programming interface (API) (or graphical user interface GUI), etc. [2]. Remote communication between the high-quality RWD and the user's host, e.g., via the Bluetooth protocol, local network (Ethernet), Wireless Local Area Network (WLAN), and others [3], is becoming a desired standard. It should be emphasized that cheap RWDs are commonly equipped only with wired serial communication ports, i.e., USB, UART, or old RS232. This significantly hinders the design of a broad but also inexpensive system based on local area networks. It turns out that with the progress of computerization and the use of computer networks is one of the cheapest solutions.

The size of the IZ in an RFID system depends primarily on the types of devices used [4]. Types of RFID systems can be divided according to the range of operation depending on the output power of the transmitter in the RWD, the construction of the RWD antenna, and the frequency band:

- Proximity range system: operation range is equal for up to several centimeters.
- Medium range system: operation range is equal for up to several dozen centimeters.
- Long range system: operation range is equal for up to several meters.

Of course, the greater the operation range, the greater the output power and sensitivity of the RF front-end in RWD, and in consequence, the device is more expensive.

A comprehensive survey of the read/write devices and the possibilities of their use in the Internet of things (IoT) applications is presented in [5]. According to the construction, authors distinguish free types of RWDs: (1) fixed or desktop devices, (2) stationary devices compacted with an antenna/antennas, and (3) portable/handheld or wearable readers. In the first two groups, the price of the product starts from 1000 USD to several thousand USD, depending on the operation range and auxiliary functions. The third group represents lightweight and cheap devices that can serve as the basis for developing a broad-range RFID system. The use of ready-made final products, available from many manufacturers (e.g., GAO RFID Inc., Ontario, Canada, or Winnix Technologies, Shenzhen, China), unfortunately, does not allow investigators to create their own solutions and implementation, especially when conducting research. Moreover, since they are equipped with full-service software, they are still more expensive than semi-finished products in the form, e.g., PCB board (e.g., CPR74 by Feig, Weilburg, Germany). Therefore, attempts are being made to create easy-made RDWs, in which researchers can freely interfere with the software code and implemented functions. In [6], the authors provide the construction of the handheld device based on a ThingMagic M6e Development Kit operating in the UHF band and communicating with the server via Wi-Fi. The developed software is written in Javascript language. The authors compare their product with other mobile devices that have some serious drawbacks, such as unmodifiable and proprietary software, limited possibilities of customization, or non-configurable hardware. The real distributed RFID system is developed in [7] on the basis of the RWD MF RC522 module by NXP Semiconductor (Eindhoven, Netherlands) and Zig Bee 2.4 GHz protocol. The authors used much more advanced means in other to create their construction; thus, the system cannot be easily multiplicated.

2. Conception of Research Platform

In the paper, the authors strive to develop the scalable platform of an automatic object identification system on the basis of a simple RWD with the low power output of RF front-end and one of the standard serial interfaces for exchanging data (e.g., CPR74 Feig HF RWD module). Instead of the sophisticated data transmission block built-in the RWD, they use an external microcontroller module (one of popular in the electronic market) with a Wi-Fi section. There is no doubt that both components (RWD and microcontroller module) have to communicate via a serial interface. Among others, the universal asynchronous/synchronous receiver/transmitter is the simplest driver (for establishing RS232 protocol and derivatives) and is commonly used in industry practice. If both devices operate at the same voltage supply (e.g., 3.3 V or 5 V), then an even voltage level converter is not necessary. When standard drivers are used, data can be sent over a distance of several meters. The components also may communicate via other available wired simple interfaces (e.g., SPI, I2C) [8]. Then, additional elements such as temperature, light, accelerometer sensor, etc., could also be added, if necessary. Additional information on the application environment could support the process of identification.

Another advantage of the small RWD node is its low energy consumption and, thus, it can be used as a portable device powered by accumulators or a renewable energy harvester. Referring to the possible hardware implementation of the described idea, the Wi-Fi ESP8266 microcontroller module (by Espressif Systems, Shanghai, China) can act as the access point in the designed platform [1,8–13]. There are also other constructions of a similar type, which are unfortunately more expensive, e.g., Arduino with Ethernet interface (by Arduino project, Ivrea, Italy) or extension modules for ARM STM32 Nucleo (ST Microelectronics, Geneva, Switzerland).

The host is the next component of the designed platform of automatic identification. An inexpensive Raspberry Pi minicomputer (by Raspberry Pi Foundation, Cambridge, GB) is a perfect choice for the server for data acquisition and system management [14]. Of course, it is possible to use more sophisticated embedded systems, e.g., Intel development boards that are based on one of field programming gate array (FPGA) chips, as well as a typical personal computer PC. Although server services can be run under any operating system or in the so-called "cloud" [3,8,12,13], the Raspberry Pi minicomputer was chosen because of its small size and price as well as easy and quick configuration to the required application.

Thus, using the mentioned three components, it is possible to create a scalable, distributed network of RWD devices in which the space of the IZ is freely shaped. In addition, any topology that is known from IT systems can be adapted.

All that remains is to choose software tools to create the utility application and the user interface. According to the analysis of the market carried out by the authors, there are many freely accessible software tools, including, e.g., Node-Red, Domoticz, LabView, and Wyliodrin Studio [2]. Using these tools, it is easy to create the utility program by combining functional blocks and their options and writing simple codes (own functions, procedure, etc.), supplementing the automatically generated template. Moreover, it is also worth choosing a data exchange protocol between the server and the node (RWD with microcontroller module) that does not cause too many problems in operation. For example, MQTT or HTTPS [2,14–16] are often used. The decisive factor in choosing a development environment is also the level of support for a given software, as well as the possible access to datasheets and library descriptions. On the other hand, it is always possible to write your own Web applications in one of the object-oriented languages (e.g., the commonly used combination: PHP + HTML + CSS) [2]. Although it took a lot more time, it would be a creative and individual solution.

3. Architecture of Research Platform

After the analysis of advantages and disadvantages was performed regarding possible constructions of the designed platform, it was assumed that the prototype system should operate on the basis of the client-server architecture (Figure 1). In the adopted concept, a Raspberry Pi minicomputer with the Raspberry Pi OS operating system, derived from the Linux family, was used as a server for data exchange/acquisition.

The Node-Red graphical programming tool (which has an idea of operation similar to the Wyliodrin Studio), installed on the server, is used to create data flows. This front end works in the Node.js run-time environment intended for programs that are written in JavaScript. The platform customization was conducted by creating our own plugins for Node-Red.

The Mosquitto (MQTT Broker) tool is responsible for communication between the Node-Red tool and the individual RWD nodes. MQTT is a friendly, lightweight protocol. All messages that were sent by client devices (ESP8266, as well as those sent to individual nodes from the Node-Red environment) are directed to the MQTT Broker (MQTT client/server). The broker intercepts these messages and provides them to be read by devices that listen to the given topic. Instructions/orders are messages sent after the "/" character, as shown in Figure 2. Additionally, it is possible to set appropriate flags in the data frame defining a username, password, quality of service, etc.

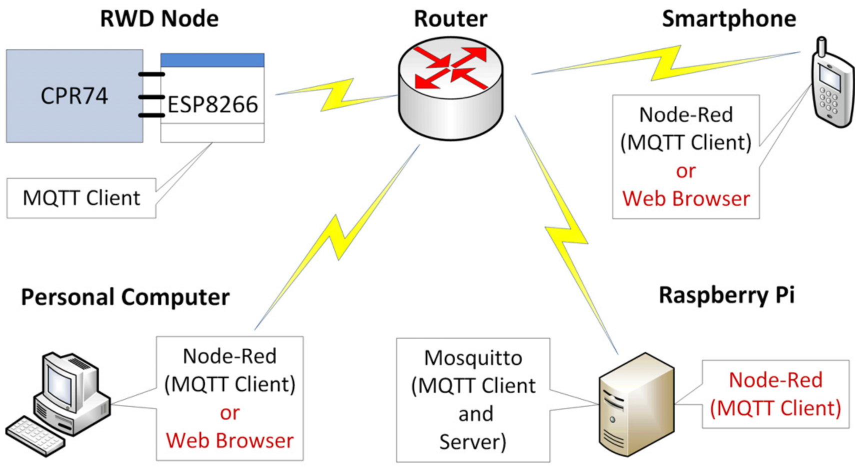


Figure 1. Architecture of development platform.

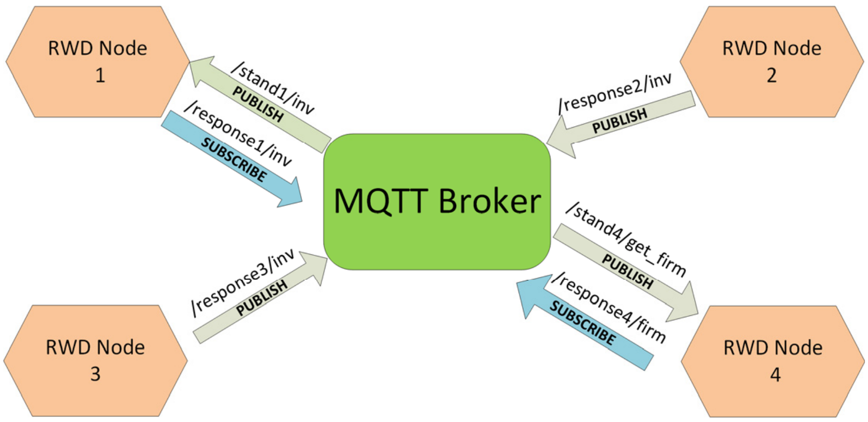


Figure 2. Idea of MQTT.

The combination of the possibilities offered by the MQTT protocol, the Node-Red tool, and the Node.js environment is a great option for creating the designed identification system. Hence, the set of these tools is chosen to implement the prototype platform.

Every RWD node is implemented on the basis of the Wemos D1 Mini development module (with ESP8266 chip) [17] and the HF band read/write device CPR74 from Feig company [18]. This particular CPR74 model was chosen because its technical and operational parameters are sufficient for the proposed system. It is equipped with a USB interface but also has a built-in UART (5 V standard). The Wemos D1 Mini transceiver provides communication between the RWD connected to it via the UART and the MQTT server via a wireless local area network. To program, the code is generated, and the Wemos D1 Mini is debugged; the Arduino IDE development environment (developer Arduino.cc, Arduino LLC, Somerville, MA, USA) with appropriate libraries were used.

It is assumed that all components of the platform communicate with the data/acquisition server using the WLAN medium by connecting to an access point that is set up in the form of a Wi-Fi router (similar architecture is often used in IoT). Each of the devices communicates with the server using the MQTT protocol. Users have access to the GUI on any device (smartphones, PCs) by entering the server's IP address and the appropriate port. The graphical interface allows users to operate the platform as well as to control the workspace of the Node-Red tool. In this way, the flow of information can be freely created. Since the Node-Red tool is installed on the server as a client of the MQTT protocol, a typical web browser can be used on end devices (smartphones, PCs) to manage information. At the stage of platform development, an option to install the Node-Red tool on each end node was also taken into account. Nevertheless, it was found that it would be unnecessary, and the Node-Red tool installed on the server was enough.

Commands to the RWD module are sent as messages from the PC to the ESP8266 transceiver in which the transcoding table is used—the message values coming from the server are assigned to the specific commands of the RWD in the form of hexadecimal instructions. The transcoding table is built on the basis of the available documentation [18], as well as using the ISOStart Feig program [19].

4. GUI of Test Platform

The main goal of the experiment is to propose a distributed RFID system based on simple RWD nodes, which could be an alternative to long-range high-power RWDs. In general, long-range devices are expensive, although very well equipped with communication interfaces, software development kits (SDK) or functional GUI, etc. So, in order to reduce costs but maintain similar functionality, it is advisable to develop the proposed platform using simple software that allows potential designers to quickly and cheaply create utility programs. It should also not be forgotten about basic security aspects, e.g., logging into the user interface.

Using Node-Red, the authors can significantly speed up the design of GUI thanks to the possibility of graphical programming in the flow editor. In Figure 3, the template of the complete program for the interface with full functionality is presented. The created RWD user interface consists of four blocks:

- Mobile menu (Figure 3a)—this block organizes the information flow between the management program and the GUI.
- Reset and clear table (Figure 3b)—this is a procedure for setting default parameters in the GUI and restarting or shutting down the Raspberry Pi minicomputer (server).
- Information subscription dashboard (Figure 3c)—this is the main utility program.
- Data publication dashboard (Figure 3d)—this is the procedure for preparing masseges and sending them to other parts of the system.

In the Information subscription dashboard, there is the Subscription section which is responsible for listening to the desired message and, after receiving it, forwarding it to the next blocks. The Functions section cyclically receives data, processes it, and passes it further. It is designed to extract specific information from the received data, create objects from them and transfer the data to the CSV converter (e.g., add separators). Then, results are saved in .txt and .csv files in the specified location. The results can also be presented in the output table of the GUI. The Background and Logo section provides access to the code of procedures responsible for the GUI appearance (CSS, HTML), indicating photos, background, etc.

In the Data publication dashboard, the Buttons section serves as the input of relevant data for the blocks from the Publication section. The received data are published as messages on a given topic to the MQTT server. The Inventory Loop block works as a switch; therefore, in the OFF state, it gives the possibility of resetting/clearing the relevant fields/indicators, whereas, in the ON state, it sends previously entered commands to the MQTT blocks.

In special blocks of the programming environment, it is possible to write some parts of the user code, as shown in Figure 4. In this way, new objects are defined, such as organization in CSV file (Figure 4a), the appearance of buttons (Figure 4b) or their functions (Figure 4c), and designer procedures (Figure 4d). For ease of use, an additional graphical interface is also created for devices with a smaller screen diagonal (e.g., mobile phones). In this part, some unnecessary elements of the user window are removed, and connecting flows are revised (Figure 4b). Nevertheless, the interaction with the main GUI is kept on the same functionality.

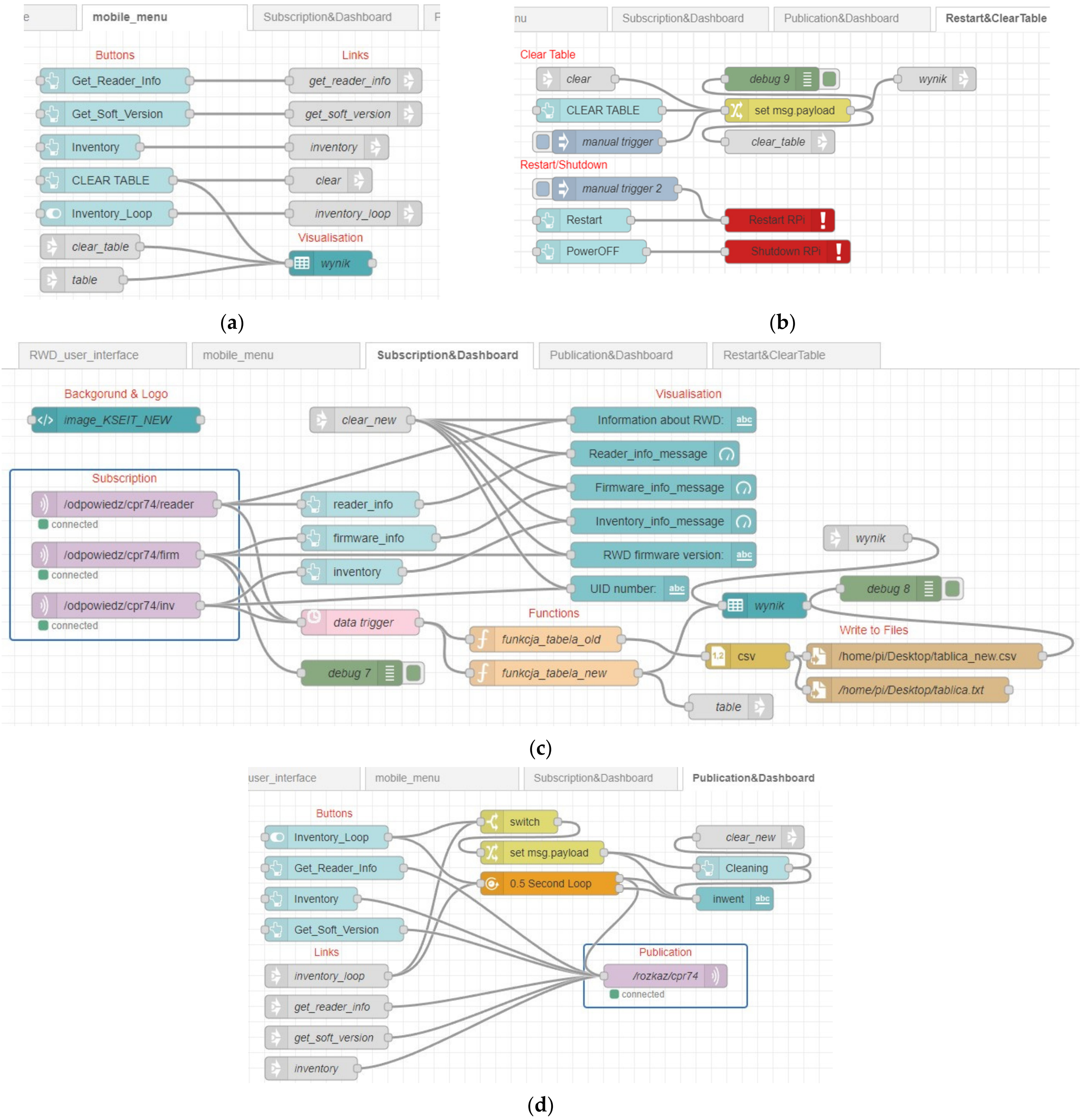


Figure 3. Block diagram of the program flows in Node-Red: (**a**) Mobile menu; (**b**) Reset and clear table; (**c**) Information subscription dashboard; (**d**) Data publication dashboard.

The Node-Red tool also provides various mechanisms for securing designed systems, i.e., login windows, the exchange of clearance certificates, etc. These mechanisms are selected adequately to the level of security that is assumed in a developed system. Since no confidential data are transferred on the test platform, only the option to log in to the workspace (the area where flows are created) is implemented. The password is encrypted with a special tool (from the terminal level) which is attached to the downloaded packages of the Node-Red library. After signing into the utility program, the client stays logged in for several days by default (the login period can be set in the configuration file).

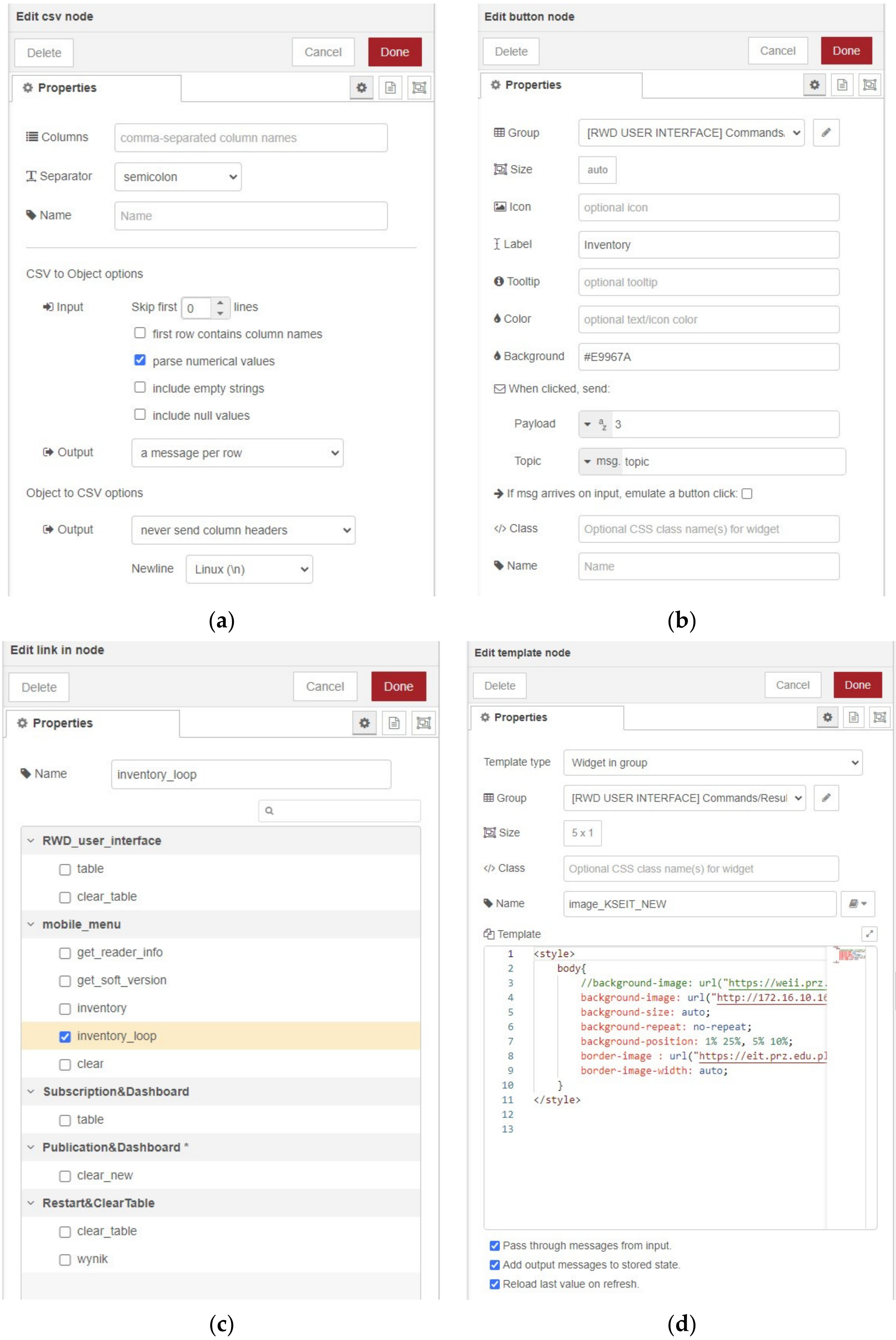


Figure 4. Designing in Node-Red—exemplary objects: (**a**) Options of CSV objects; (**b**) Button appearance settings; (**c**) Button function assignments; (**d**) Designer procedure.

In the Node-Red editor, it is possible to easily configure the GUI layout (background, additional logo, etc.). As an example, the user interface conception by the authors is shown in Figure 5. In the layout, the sub-windows are used. The launched tabs can be treated as a separate GUI, and they can be bounded with various results. The reduced interface for mobile devices provides only basic functions, and the size of the window components is adjusted to smaller resolutions of screens.

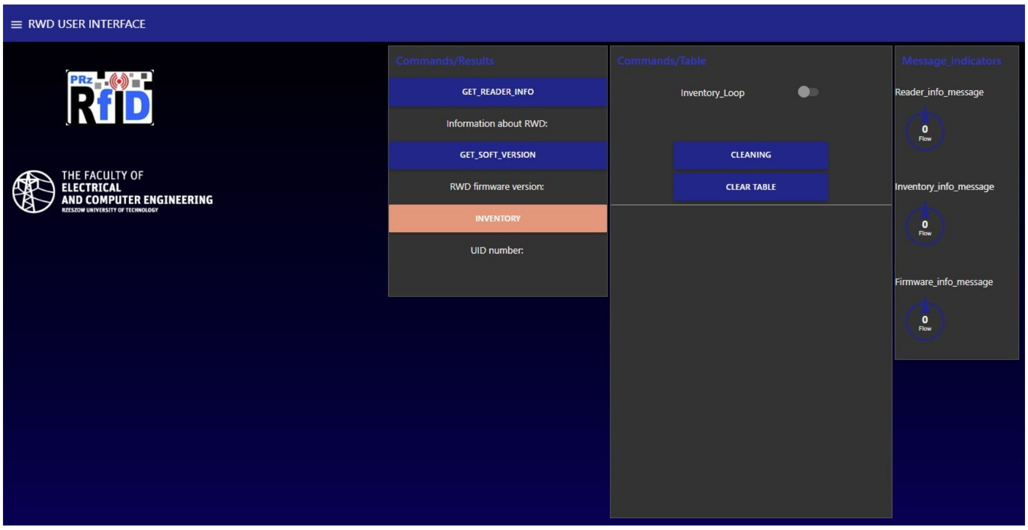
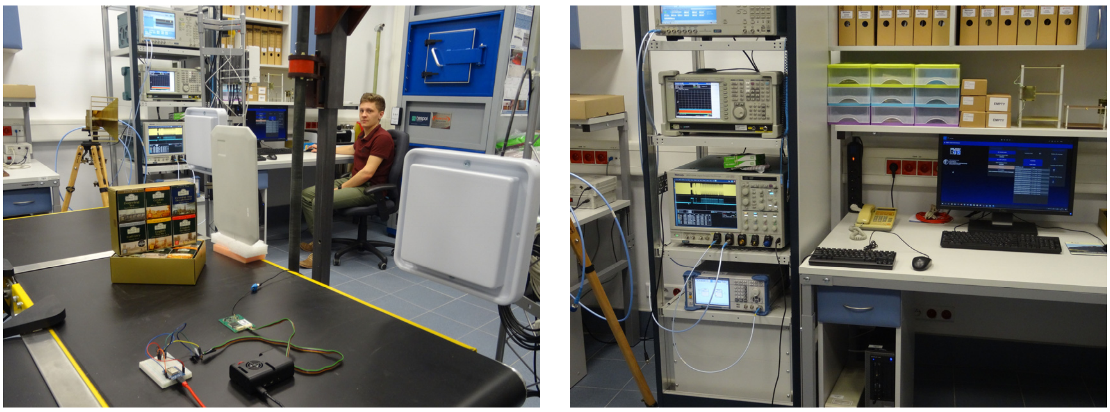


Figure 5. GUI conception by the authors.

The server can be turned off remotely after finishing work or can be restarted in any case of problems with its operation. These options help operators to manage the system and are useful when the users have lost access to the monitor or other input/output devices connected to the Raspberry Pi minicomputer. It also prevents the sudden disconnection of the power supply. The Raspberry Pi minicomputer restarts automatically when the power is reconnected.

5. Test Platform Evaluation

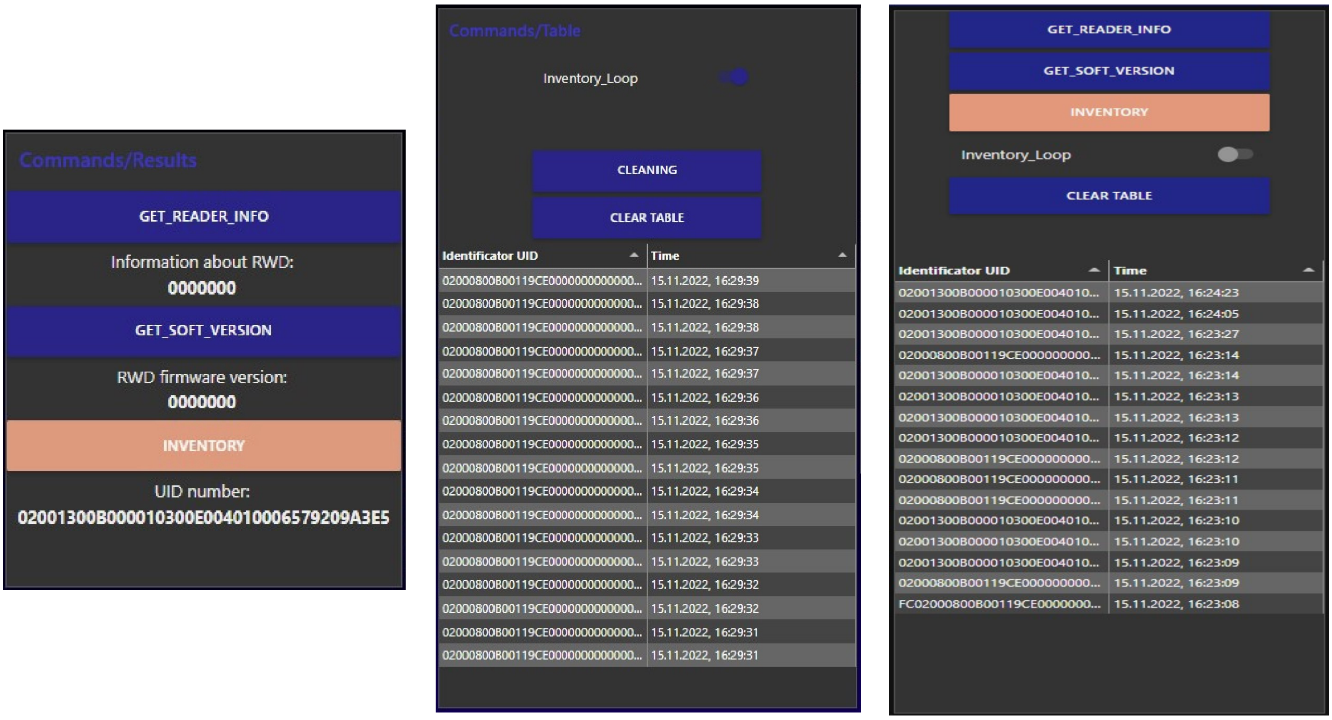
The basic tests of the developed platform were carried out using commonly available HF transponders. The experimental laboratory stand is shown in Figure 6. It includes the server, transceivers connected to the RWDs, RFID tags, antenna for the HF band, the PC computer, and the access point of the local wireless network.



(**a**) (**b**)

Figure 6. Laboratory stand: (**a**) Operational space of RFID system; (**b**) Measurement center.

The basic tests that were performed were aimed at checking the correct operation of the created system and its effectiveness. First, as part of the communication tests, it was checked whether the system could correctly read the information about the RWD (Figure 7a).



(**a**) (**b**) (**c**)

Figure 7. Tests: (**a**) Retrieving RWD system information; (**b**) Identifying RFID objects; (**c**) GUI for mobile devices.

The attempts to identify various groups of transponders were performed at an axisymmetric alignment of their antennas and the RWD antenna in short time intervals (Figure 7b). The tests showed 100% efficiency of the identification. The same tests were performed for the GUI dedicated to mobile devices (Figure 7c).

The effectiveness of the tag identification system depends on the quality of the control program installed on the transceiver. If the program is written incorrectly, the delay in object detection could be too long. The microcontroller has to simultaneously send orders to the RWD, receive responses, and then send information to the MQTT server and further to the MQTT client, which is the Node-Red tool. However, as the tests show, the results are satisfactory.

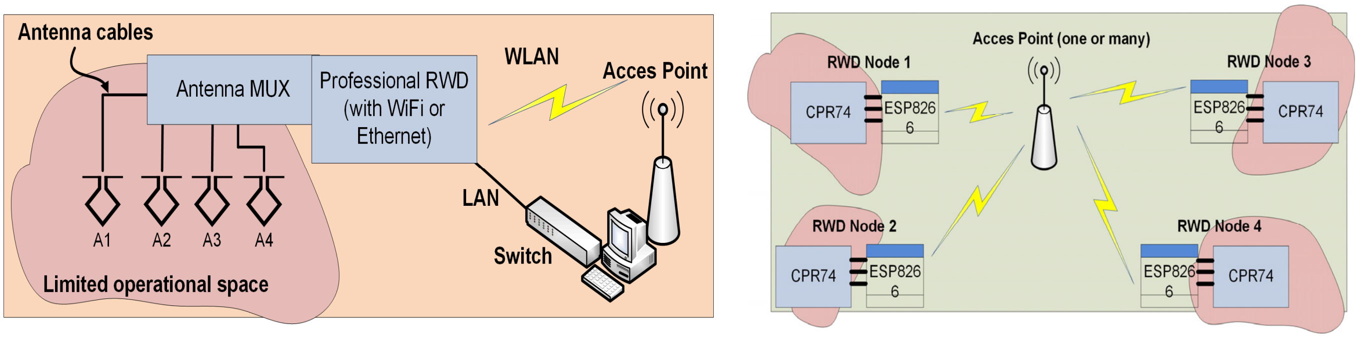
In order to enable the review of the entire identification process, the possibility of saving data to a file with the ".txt" extension is provided. The gathered information can be successfully saved to databases or passed to other programs.

6. Distributed Platform Advantages

The functionalities that are presented on the prototype platform are only a limited demonstration of the potential that lies in the used tools and is intended to show an easy way to build similar stands/systems at low costs. The system can be freely expanded and supplemented with various functionalities. The only limitation is the imagination of a potential designer.

The typical arrangement of RFID systems with the long-range HF RWD is shown in Figure 8a. For example, it can be built on the basis of Feig LRM2500-A RWD, which is equipped with:

- Protocols ISO/IEC 15693, ISO/IEC 18000-3
- Communication interfaces USB/LAN/RS232/RS485/WLAN
- The output power of the transmitter up to 12 W



(**a**) (**b**)

Figure 8. RFID system: (**a**) Standard arrangement; (**b**) Distributed platform.

In order to achieve the large IZ, it is necessary to use the antenna multiplexer (e.g., 4-channel multiplexer Feig ANT.MUX.M4) with compatible antennas (e.g., Feig ANT310/310-A). This bundle provides approximately 60–70 cm of operational range in four directions depending on the position and rotation of each antenna. The cost of the above-mentioned implementation is about 7000 USD (Table 1).

Table 1. Comparison of RFID systems.

Parameter	Standard Arrangement	Distributed Platform
Price	7000 USD	700 USD
Operational range at one antenna	About 60–70 cm	About 20–30 cm
Output power of RF front-end	12 W	450 mW
Possibility of shaping IZ	Limited by length of signal cables	No limited Depends on the number of cheap nodes. High mobility of nodes
Ability to development in easy and fast way	Highly advanced RWD and control software	Greater availability of free development tools Nodes can be supplemented with, e.g., sensors on the local bus
Environmentally friendly	No High power supply There is a risk to exceed norms of the electromagnetic field *	Yes Possible battery supply or energy harvester

* The limits of the electromagnetic field strength for the HF band are specified in ETSI EN 300 330—the problem is described, e.g., in [4].

A ten times cheaper system can be built using the proposed distributed platform. A set of devices for assembling the proximity/medium range of HF RFID systems contains, e.g., Feig CPR74 RWD:

- For protocols ISO/IEC 14443, ISO/IEC 15693, ISO/IEC 18000-3;
- With communication interfaces USB and RS232 (TTL);
- Up to 450 mW of the output power of the transmitter;
- One antenna on board.

The RWD can cooperate with an expander module of WLAN, e.g., ESP8266 Wemos D1 mini that is equipped with:

- Wi-Fi wireless communication interface in 802.11 b/g/n standard;
- Converter USB-to-RS232 (TTL);
- Supply voltage of 3.3 V or 5 V;
- General input/output ports (GPIO) and serial buses in I2C, SPI or 1-Wire standard;
- Analog-to-digital converter (ADC).

The system can be supplemented with an external antenna (e.g., Feig ANT100/100-U.FL-A) and also, but not necessarily, a multiplexer (e.g., Feig 4-channel CPR.ANT.MUX.M4). This bundle provides approximately 20–30 cm of operational range in each direction.

The RFID devices have been selected from the portfolio of only one company due to comparative purposes. Nevertheless, there are other component substitutes for the proposed distributed application. For example, instead of ESP8266, the following modules can be used as the Ethernet/UART converter (by WIZet, Santa Clara, CA, United States):

- Wiz145SR module—dedicated GUI is provided for module configuration; the price is about five times lower (up to 50 USD)
- WizFi210 module—with an additional interface of the SPI standard, the producer provides a list of WizFi310 AT commands that can be inputted through the UART line; the price can be up to 10 USD.
- Similarly, instead of the Raspberry Pi microcomputer, the other embedded platforms can be used, e.g., ARM STM32, ARDUINO, or one of the relatively expensive FPGA boards (e.g., by Intel, Santa Clara, CA, USA)

Thus, considering the economic aspects of developing the distributed identification system, it is necessary to take into account the costs of RWD nodes. When the proximity/medium range was selected, the price of the RWD node would be about 700 USD, whereas, in the case of long-range equipment, it would be about 10 times higher.

7. Summary

As was proved in this paper, the distributed system of autonomous identification with the large IZ can be easily designed using simple RWDs and some kind of transceiver. Moreover, it can be performed cheaply and quickly using commonly available software tools. As shown, there is no need to buy expensive RWDs and antenna multiplexers. Since the price of the used components (cheap RWD and Wi-Fi minimodule) is very low, it is possible to use a larger number of RWD nodes so as to cover the largest possible operational space. In addition, such a nod has no limitations to adding useful sensors to it or other input/output devices.

Another important thing is the mobility of such a construction. RWD nodes can be located anywhere. The only barrier is the availability of a connection to a wireless network, WLAN. Each of the modules, including the server, can be powered by a battery or other sources of energy harvested from the environment. This opens up new possibilities for the quick configuration of a distributed utility system or research stand. The server, on the other hand, can be located anywhere as long as it is visible in the WLAN for each RWD system with a Wi-Fi module. Despite its simplicity, the system can work in a variety of conditions.

In this way, it is possible to create a platform for IoT. The authors, considering the platform based on the HF RFID system, have focused rather on the idea of a distributed system of object identification than on technical means. The platform could just as well operate based on completely different hardware/software tools, using different data transfer protocols, etc. The proposed concept can also be expanded for RFID systems working in the UHF band.

Author Contributions: Conceptualization, P.J.-M.; methodology, P.J.-M.; software, B.W.; validation, P.J.-M. and M.W.; formal analysis, B.W.; investigation, B.W.; resources, B.W.; data curation, B.W.; writing—original draft, B.W. and M.W.; writing—review and editing, M.W.; visualization, B.W. and M.W.; supervision, P.J.-M.; project administration, M.W.; funding acquisition, P.J.-M. and M.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research paper was developed under the project financed by the Minister of Education and Science of the Republic of Poland within the "Regional Initiative of Excellence" program for years 2019–2023. Project number 027/RID/2018/19, amount granted 11 999 900 PLN.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All calculated and measured data will be provided upon request to the correspondent authors by email with appropriate justification.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Finkenzeller, K. *RFID Handbook—Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication*, 3rd ed.; Wiley: Hoboken, NJ, USA, 2010.
2. Guinard, D.D.; Trifa, V.M. *Building the Web of Things: With Examples in Node.js and Raspberry Pi*, 1st ed.; Manning Publications Co.: Shelter Island, NY, USA, 2016.
3. Laxmi, A.R.; Mishra, A. RFID based Logistic Management System using Internet of Things (IoT). In Proceedings of the 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 29–31 March 2018.
4. Jankowski-Mihułowicz, P.; Węglarski, M. Definition, Characteristics and Determining Parameters of Antennas in Terms of Synthesizing the Interrogation Zone in RFID Systems. In *Radio Frequency Identification*; Crepaldi, P.C., Pimenta, T.C., Eds.; INTECH: London, UK, 2017; Chapter 5; pp. 65–119.
5. Costa, F.; Genovesi, S.; Borgese, M.; Michel, A.; Dicandia, F.A.; Manara, G. A Review of RFID Sensors, the New Frontier of Internet of Things. *Sensors* **2021**, *21*, 3138. [CrossRef] [PubMed]
6. Bunker, R.; Elsherbeni, A. A Modular Integrated RFID System for Inventory Control Applications. *Electronics* **2017**, *6*, 9. [CrossRef]
7. Ji, W.; Li, L.; Zhou, W. Design and Implementation of a RFID Reader/Router in RFID-WSN Hybrid System. *Future Internet* **2018**, *10*, 106. [CrossRef]
8. Kodali, R.K.; Anjum, A. IoT Based Home Automation Using Node-RED. In Proceedings of the 2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT), Bangalore, India, 6–18 August 2018.
9. Chanthakit, S.; Rattanapoka, C. MQTT Based Air Quality Monitoring System using NodeMCU and Node-RED. In Proceedings of the Seventh ICT International Student Project Conference, Nakhonpathom, Thailand, 11–13 July 2018.
10. Macheso, P.; Manda, T.D.; Chisale, S.; Dzupire, N.; Mlatho, J.; Mukanyiligira, D. Design of ESP8266 Smart Home Using MQTT and Node-RED. In Proceedings of the International Conference on Artificial Intelligence and Smart Systems (ICAIS-2021), Coimbatore, India, 25–27 March 2021.
11. Saxena, A.; Tyagi, M.; Singh, P. Digital Outing System Using RFID And Raspberry Pi with MQTT Protocol. In Proceedings of the 2018 3rd International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Bhimtal, India, 23–24 February 2018.
12. Rajalakshmi, A.; Shahnasser, H. Internet of Things using Node-Red and Alexa. In Proceedings of the 2017 17th International Symposium on Communications and Information Technologies (ISCIT), Cairns, Australia, 25–27 September 2017.
13. Zare, A.; Iqbal, M.T. Low-Cost ESP32, Raspberry Pi, Node-Red, and MQTT Protocol Based SCADA System. In Proceedings of the 2020 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), Vancouver, Canada, 9–12 September 2020.
14. Culic, I.; Radovici, A.; Rusu, C. *Commercial and Industrial Internet of Things Applications with the Raspberry Pi*, 1st ed.; APress C.A: Berkeley, CA, USA, 2020.
15. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]
16. Hue, A.; Sharma, G.; Dricot, J.-M. Privacy-Enhanced MQTT Protocol for Massive IoT. *Electronics* **2022**, *11*, 70. [CrossRef]
17. Espressif Systems, ESP8266EX. Datasheet, Version 6.8. 2022. Available online: www.espressif.com (accessed on 1 December 2022).
18. Feig Electronics, ID CPR74 RFID Reader Module, Feig Electronic GMBH, Weilburg, Germany. Available online: www.feig.de (accessed on 1 December 2022).

19. Feig Electronics, ID ISOStart, V7.02, H81291-8e-ID.doc, 2006, Feig Electronic GMBH, Weilburg, Germany. Available online: www.feig.de (accessed on 1 December 2022).