PROJECT, DESIGN AND MANAGEMENT

ISSN: 2683-1597

PROJECT DESIGN & MANAGEMENT

How to cite this article:

Toledo, M. A. (2022). Seguimiento y control de ganado vacuno mediante geoposicionamiento. *Project, Design and Management,4*(1), 91-113. doi: 10.35992/pdm.4vi1.828

MONITORING AND CONTROL OF CATTLE BY GEOPOSITIONING

Marcelo Alejandro Toledo
Universidad de la Cuenca del Plata (Argentina)
toledomarcelo_for@ucp.edu.ar · https://orcid.org/0000-0001-6389-1590

Abstract. The development of this research work called *monitoring and control of cattle by geopositioning*, whose priority line is innovation and technology and is of the applied type. With it, it will be possible to develop a prototype that will allow determining the location of the cattle in real time. It allows receiving location coordinates in real time and then sending them through the cellular network to an Internet of Things web platform that operates with this information and feeds different widgets in which the location of the cattle can be observed. To achieve this, a Global Positioning System is used, whose objective is to determine the spatial coordinates of points with respect to a world reference system. The system is based on the Arduino board connected to a SIM808 module that offers GPS technology and allows the satellites to request positioning coordinates. In the future, it is estimated that all farmers must have an animal control system, because it is essential to monitor and control in real time, to know the location, combat theft and control the health of the same. The methodology used for the development of the system is that of prototyping.

Keywords: Geopositioning, IoT, Cattle, Prototype.

SEGUIMIENTO Y CONTROL DE GANADO VACUNO MEDIANTE GEOPOSICIONAMIENTO

Resumen. El desarrollo del presente trabajo de investigación denominado *seguimiento y control de ganado vacuno mediante geoposicionamiento*, cuya línea prioritaria es innovación y tecnología y es del tipo aplicada. Con el mismo se logró desarrollar, un prototipo que permite determinar la ubicación en tiempo real el ganado vacuno. Permite recibir coordenadas de ubicación en tiempo real para luego enviarlas mediante la red celular a una plataforma web de Internet de las Cosas que opera con esta información y alimenta diferentes widgets en los cuales se puede observar la ubicación del ganado. Para lograr esto se utiliza un Sistema de Posicionamiento Global, cuyo objetivo es determinar las coordenadas espaciales de puntos respecto de un sistema de referencia mundial. El sistema está basado en la placa Arduino conectado a un módulo SIM808 que ofrece la tecnología GPS y permite solicitar a los satélites coordenadas de posicionamiento. En un futuro, se estima, que todas las ganaderas deberían contar con un sistema de control de animales, debido a que es primordial realizar seguimiento y control en tiempo real, para conocer la ubicación, combatir el robo y controlar la salud del mismo. La metodología utilizada para el desarrollo del sistema es la del prototipado.

Palabras clave: Geoposicionamiento, IoT, Ganado, Prototipo.

Introduction

Argentina has historically been an agricultural-livestock farming country and, given its size and production volumes, it is considered one of the world's major food suppliers. In recent years, among other things, due to commodity prices (firstly, the price of soybeans and secondly, the price of wheat) and political factors, livestock farming has lost ground to agriculture. (Agrositio, 2008)

The Argentine northeast (NEA) is made up of the provinces of Corrientes, Misiones, Chaco, and Formosa in their entirety and part of the provinces of Entre Ríos and Santa Fe, which share agroeconomic characteristics that give the region its own identity. It has 19.46 million heads of cattle, which represent 40% of the national cattle stock, according to the first vaccination campaign of 2010 by SENASA. In order of importance, Santa Fe has 6.03 million heads, Corrientes 4.87 million, Entre Ríos 3.98 million, Chaco 2.38 million, Formosa 1.79 million, and finally Misiones only 410 thousand heads. (Acosta, F. and others, 2012).

Livestock producers are grouped into strata according to the amount of cattle they own, according to SENASA (2010). In this way, the provinces of Santa Fe with 44%, Misiones with 35% of cattle owned by the smallest producers (owning up to 100 heads), and Corrientes with 19% of cattle owned by the largest establishments (owning more than 5,000 heads).

According to INTA (2015), there is clear evidence that livestock production in the Province of Formosa has grown quantitatively in recent decades. There are several causes that drove this local productive intensification. In particular, the process of agriculturization that the country has been undergoing for more than a decade, which led to the growth of livestock in regions such as the NOA and NEA. Thus, livestock farming in Formosa, as in the rest of the country, has undergone structural and geographical changes as a consequence of the transformation of the agricultural sector.

In a long-term view, regardless of short-term cycles, two stages can be identified in the evolution of the cattle stock in Formosa during the last 12 years. The first one between 2003 and 2008 in which there was a significant increase in the stock, which expanded at a cumulative annual rate of 5%. And in the second stage between 2009 and 2015, there was no notable rate of change in the stock, but after a slight drop in the stock, a new phase of retention began to the point that in 2013 it reached levels higher than those recorded in 2008. Therefore, it could be argued that the minimum values reached during 2008 constituted a floor from which the Formoseño livestock cycle evolved in recent years; the data released by INTA in 2015.

However, "cattle rustling" continues to be in our country one of the scourges that has hit the livestock sector the most, known in the country jargon as "cattle rustling"; it is a criminal behavior difficult to control. To this, we must add the clandestine slaughter that implies a risk to human and animal health, with the corresponding tax evasion, transport, and marketing without compliance with sanitary and bromatological standards, seriously compromising the human and animal health aspect, with serious risks of the spread of various diseases.

This criminal activity is favored by the large rural extensions of our country and our province, which often prevent the prevention of an adequate control in the care of the property of livestock, cattle, etc., added to the deficiencies in the infrastructure of the rural police of the country and the legislation that ordered a greater repressive action, established in Law No. 25,890 of the Criminal Code of the Province of Formosa.

The loss of cattle by cattle farmers in the Province of Formosa is a very important issue to solve, currently the same have no way to effectively control the location of their animals and can only hire people to take care of their cattle, but that implies a high economic cost. (AM990formosa, 2017)

According to the president of the Formosa Rural Society, Carlos Montoya, cattle rustling has increased in the last year in the border zone, and he expressed the concern of the farmers about the violence and aggressiveness of the rustlers. (Agrositio, 2008)

This is evidenced in the note made to a cattle rancher located to the north of the capital of Formosa, who commented with much anguish, the serious damage caused to the economy and work, the sustained and increasingly sophisticated action of cattle rustlers. Cattle rustling depredates the livestock economy of the province and hacks the labor source. (Panaroma Regional, 2018)

According to members of the Livestock Association of Formosa, 80% of animal losses in the Province of Formosa are due to livestock theft, 15% to strays in other fields, and 5% to diseases; for this reason, they proposed the feasibility of using ICTs to control and know the location of their animals at a given time.

Nowadays, technologies are migrating to all fields of the economy, providing innovation and constant progress. More and more we find ourselves in a highly interconnected environment, where the goal is the development and improvement of current systems, seeking greater viability and feasibility in the processes.

From this need arises the idea of the present research project in order to satisfy this basic need for farmers using new technologies combining the concepts of Hardware and Free Software, Internet of Things, and satellite geopositioning.

Based on the above, a prototype was developed to determine the real-time location of cattle. The system allows to receive location coordinates in real time and then send them via satellite signal to a web platform of Internet of Things (IOT) that operates with this information and feeds different widgets in which you can see the location of cattle, date and time of the same.

To achieve this, a global positioning system (GPS) will be used, whose objective is to determine the spatial coordinates of points with respect to a world reference system.

According to the Bankinter Foundation of Innovation, the Internet of Things or IoT is based on interlinking everyday objects with the Internet, thus making it easier to obtain and manage information. This is what the well-known sensor networks consist of, which are made up of a specific number of devices and whose data goes to a coordinating node or router and from there to a web server where it is possible to store and maintain the most relevant information.

This type of technologies is not only being seen in the field of intercommunication and the military; it is also being seen how this type of aspects have migrated to sectors in which very little was believed to be necessary, such as the field of agronomy in all its subdivisions. This field still continues to be managed with little technological affection in the Province of Formosa.

The general objective of the research was to determine the applicability of a technological system of free hardware and software for the monitoring of cattle in the province of Formosa in the period 2020 - 2021.

And its specific objectives were:

- Analyze alternatives for the control and monitoring of cattle location.
- Develop the prototype and interface for the tracking system.
- Simulate the circuit of the device to verify the proper functioning of the prototype.

Frame of reference

For the elaboration of this project, a compilation of several sources of bibliographic research has been carried out, such as digital repositories of several universities, scientific articles, scientific journals, etc., in search of topics similar to the proposed one, which have developed works that are essential for the elaboration of this research.

In 2013, Ángel René Canché UC and Jonathan Ismael Mukul Chi developed a "LocaPet. Satellite Locator for Pets," where they describe the development of a system that is responsible for monitoring a pet within a specific area. The hardware used in this system consists of an Arduino UNO 328 board model MIC-06664, GPS GY-GPS6MV1 with GPS NEO-6M chip, GSM/GPRS SIM900 board model WGW-06633. The software used is open source using the Arduino platform and AT codes. This system issues a text message alert with the geographic location, time and date that is sent to the owner in case the pet leaves the indicated area for a longer time than the established time, which allows to know the location of the pet anywhere as long as there is GSM cellular network coverage.

In 2015, Ezequiel Gorandi, Nicolás Clemares, and Andrés Moltoni developed a "Collar with GPS technology for animal monitoring," where such system allows obtaining the instantaneous coordinates of the position and communicates through the NMEA 0183 protocol, with an 8-bit microcontroller to process and store the information in non-volatile memory. the programming language used is Python and such information is uploaded to a server and can be accessed through a web application in which the times between samples can be modified. This system was used in cattle, goats, sheep, and also in sheepdogs.

In 2016, Carlos Andrade Parreño developed the "Design and implementation of a client-server system for sending position and vital signs of pets on mobile devices on the Arduino platform." This project is based on the development of a prototype that prevents the loss of pets, sending vital signs and location through a hardware that is located on the chest of the animal with the help of a harness; this sends information to a web server and database, which are processed and stored for display on a mobile device or web interface. The hardware used consists of an Arduino Uno board, SIM Module 908C, Accelerometer Transducer Module MMA7361, Infrared Temperature Sensor MLX90614. The software for the development of the Android application is done using Java language and, in certain parts, makes use of XML language, Arduino language.

In 2016, Guido Buscetti Castro, Matías Prieto, Joaquín Muguerza, and Martín Ríos resumed a project they started at the university: "Do something to prevent cattle theft," investigating they came to the conclusion that it was necessary to know "what was happening to the animal when it is about to get sick or when it is going into rutting, all very important data for the producer." Late last year, they designed a working prototype of a collar that collects data and began testing it in partnership with the Veterinary School of the Universidad Nacional de La Plata. "The hardware connects to the Internet and sends the information to a platform capable of alerting the producer when something happens to the animal that requires action on his part."

The development is important because it allows, through early detection, to help the producer to save a lot of money in drugs or even to avoid the slaughter of the animal. In addition, through the detection of estrus, whose alert is sent via SMS, the work of the field person is simplified and allows optimizing the short window of time to inseminate the animals.

The unstoppable advance of the Internet has had a radical impact on the economy and society of the 21st century. The convergence of Information and Communication Technologies has transformed the traditional Internet into the Internet of Things (IoT), to such an extent that it has managed to insert itself into the productive system. The result in industry has been a new organizational model and a disruptive change that has been called the "fourth industrial revolution" or "Industry 4.0"; a term coined by the German government to describe the Smart Factory, i.e., "a vision of computerized manufacturing with all processes interconnected by IoT" (Romero et al., 2017).

Based on the problem of the loss of cattle by farmers in the Province of Formosa, which needs a prompt response, this research aims to provide a quick solution through an electronic device located in the caravans of cows, using the concepts of the Internet of Things, geopositioning, and hardware technologies and free software.

According to Karen Rose, Scott Eldridge, Lyman Chapin (2015), the concept of the Internet of Things is an emerging topic of technical, social, and economic importance. Consumer products, durable goods, cars and trucks, industrial and utility components, sensors, and other everyday objects are now being combined with Internet connectivity and powerful data analytics capabilities that promise to transform the way we work, live, and play. Projections of the impact of the IoT on the Internet and the economy are impressive: some anticipate that by 2025 there will be as many as 100 billion IoT-connected devices.

In general, the term Internet of Things refers to scenarios in which network connectivity and computing power are extended to objects, sensors, and everyday items that are not typically considered computers, allowing these devices to generate, exchange, and consume data with minimal human intervention. However, there is no single, universal definition. (Karen Rose, Scott Eldridge, Lyman Chapin, 2015)

According to Ghio M. Gina (2008), the Global Positioning System or GPS, although its correct name is NAVSTAR-GPS1, is a global satellite navigation system that allows the position of an object, person, vehicle, or ship to be determined worldwide. We can reach an accuracy of up to centimeters, using differential GPS, but the usual is a few meters.

GPS operates through a network of 27 satellites (24 operational and 3 backup satellites) in orbit, 20,200 km above the globe, with synchronized trajectories to cover the entire surface of the Earth. When a position is to be determined, the receiver used for this purpose automatically locates at least three satellites in the network, from which it receives signals indicating the position and clock of each of them. Based on these signals, the device synchronizes the Positioning System clock and calculates the delay of the signals, i.e. the distance to the satellite. By "triangulation," the three satellites calculate the position where the GPS is located. Triangulation, in the case of the Global Positioning System, is based on determining the distance of each satellite from the measuring point. Knowing the distances, the relative position with respect to the three satellites is easily determined. Knowing also the coordinates or position of each one of them by the signal they emit, the absolute position or real coordinates of the point and measurement is obtained. An extreme accuracy is also achieved in the GPS clock, similar to that of the atomic clocks carried on board each of the satellites. (Ghio M. Gina, 2008).

When we talk about free hardware, we refer to the freedom that exists when using any device along with all its documentation. According to Delgado (2018), hardware is considered free when all the information of its hardware, designs, measurements, and tools used for the creation of such devices are shared publicly, thus helping developers so that they can improve designs and contribute much more to this type of projects. (Delgado, 2018).

Free software refers to the freedom of users to modify, copy, run, study, distribute, and improve the software. However, according to the Free Software Foundation "*a program is considered free software if users have access to the four essential freedoms, such as:*

- Freedom to run the program as the user wishes.
- Freedom to study how the program works and modify it according to the user's needs.
- Freedom to distribute copies of the software to others.
- Freedom to distribute copies of improved versions from third parties."

Methodology

Based on the aforementioned concepts, the project was divided into stages taking into account the objectives proposed at present.

The Prototype Model will be used for the design of the device and the development of the software, and the Kanban agile methodology will be used for the fulfillment of the stages of the project.

According to Sommerville (2011): "*System prototyping, where a version of the system or a part of the system is quickly developed to test customer requirements and the feasibility of some design decisions. This supports the avoidance of change by allowing users to experiment with the system before delivery to refine their requirements. As a result, the number of post-delivery requirements change proposals is likely to be reduced.* " (Sommerville, 2011, p. 44)

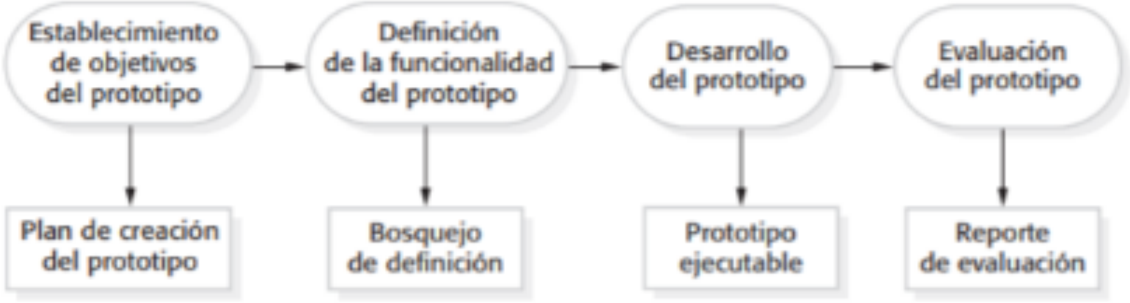


Figure 1. Prototype development process
Note: Source: (Sommerville, 2005, p. 375)

During the development, the 808 SIM module was assembled to the Arduino board, and then the programming of these boards was carried out. Different tests were carried out until the system worked correctly.

The final stage was the evaluation of the prototype, where the objectives established at the beginning were used to obtain an evaluation plan; this plan consisted of the following:

- Perform animal walk tests with the GPS configured and working.
- Control through the IoT platform the data sent by our prototype in real time.
- Once the run has been completed, check the values obtained in order to define the prototype performance and discover errors and omissions in the prototype.

System Definition and Development

In this stage, the first step was the definition of the requirements and the risk analysis, then it is explained how the selection of the motherboard and the modules necessary to carry out the development of the prototype and achieve that it meets the objectives of the project was carried out, the characteristics of the IDE for software development, necessary to carry out the programming of the motherboard, are detailed.

The system proposed in this research work is based on a prototype developed in free hardware and software, more precisely based on the Arduino UNO board as the central core, which will work together with external modules such as the SIM808 shield, which will allow, from the integrated functionalities of GSM, GPRS, and GPS, to obtain the necessary data to carry out the periodic identification of the location of a fleet of vehicles, to see the routes taken by them, and to know in real time their location, direction, and speed.

System requirements

Regarding the elicitation of requirements, these were established from interviews, but since this is a project that uses the prototyping methodology, these requirements can be modified at any stage of development, as well as new ones can be added.

The use of this methodology allows to present to the users progress in a way that allows them to see the operation and its implication in the required activities, as well as to find strong and weak aspects in the system since they are the ones who are going to use it on a daily basis.

The initial requirements obtained from the analysis of the information obtained in the interview for the system are detailed below:

- The GPS system must periodically request information from the satellites about their positional coordinates.
- The data delivered by the satellite must be sent to a web server.
- The data sent must be used by a web platform in order to represent the real position of the cell phone on the map.

These requirements were modified by testing the prototype and the following changes were obtained:

- The GPS system must periodically request information from the satellites, such as positioning coordinates (longitude, latitude, altitude), direction, speed, date, and time.
- The data sent must be used by an IoT platform, which can process the data and display its values in different widget (add to glossary).
- This platform must be able to show a history of the routes taken.

Choosing the right hardware

Different alternatives were analyzed, which are offered by the market in relation to the free programming hardware. These options were compared taking into account their characteristics and performance, with respect to the needs presented by this project.

Different alternatives in free hardware technologies

For the selection and final choice of the hardware used for this project, the main characteristics of different free hardware currently on the market were investigated, taking into account the components, their connections, their performance, and cost.

The following are the different boards analyzed with their corresponding characteristics, functionalities, and market value:

Raspberry pi

Raspberry Pi is a miniature marvel, containing within it significant computing power in a size no larger than a credit card. (Upton, Halfacree, 2016, p. 2)

The processor inside the Raspberry Pi is a Broadcom BCM2835 system-on-chip (SoC) multimedia processor. This means that most of the system components, including the central processing unit and graphics unit along with audio and communications hardware, are integrated within that single hidden component located just below the 256 MB memory chip in the center of the board. (Upton, Halfacree, 2016, p. 6)

It is not just the design of the SoC that makes the BCM2835 different from the processor in your PC or laptop, what also makes it different is that it uses a different Instruction Set Architecture (ISA), known as ARM. (Upton, Halfacree, 2016, p. 6)

Developed by Acorn Computers years ago in the late 1980s, the ARM architecture is relatively little known in the desktop world. Where it excels, however, is in mobile devices: the phone in your pocket is almost certain to have at least one ARM-based processing core hidden inside.

The combination of RISC (Simple Reduced Instruction Set) architecture and its low power consumption make it the perfect choice against desktop computer chips that demand high power consumption and CISC (Complex Instruction Set) architectures. (Upton, Halfacree, 2016, p. 6)

This, however, means that Raspberry Pi is not compatible with traditional PC software. Most software for desktop and laptop computers is built with the x86 instruction set architecture in mind, present in processors such as AMD, Intel, and VIA. Consequently, this software does not run on the Raspberry Pi, which is based on the ARM architecture. (Upton, Halfacree, 2016, p. 7)

Raspberry Pi is designed to run the operating system called GNU/Linux. Unlike Windows or OS X, Linux is open source, this means that it is possible to download the complete source code of the operating system and make any changes you want, nothing is hidden, and all changes made are publicly visible.

This spirit of open-source development has allowed Linux to quickly be modified to be able to run on the Raspberry Pi, a process known as porting, several versions of Linux (known as distributions) have been ported to the Raspberry Pi's BCM2835 chip, including Debian, Fedora Remix, and Arch Linux. (Upton, Halfacree, 2016, p. 7)

BeagleBoard

Beagles are small open hardware, open software computers that can be connected to whatever you have around the house. (https://uk.farnell.com/b/beagleboard)

Beagles mean big functionality in small packages because these small PCs can be used for all kinds of applications and can handle many of the same tasks as a desktop PC. (https://uk.farnell.com/b/beagleboard)

They are tiny, affordable, and open source for Android, Ubuntu, and different versions of Linux at your fingertips, of high performance and low power consumption.

The Beagle family's primary goal is to help students learn programming and developers to produce faster without excessive noise and expense. (https://uk.farnell.com/b/beagleboard)

The main characteristics of this plate are as follows:

The BeagleBoard is USB powered and features a 720 MHz ARM Cortex-A8 OMAP3530 processor, NEON and VFP extensions for additional acceleration, high resolution video, and the ability to stream with a portable media player, allowing you to work with all the functionality of a laptop in one small package. (https://beagleboard.org/beagleboard)

The variety of connectivity presented by the board is as follows:

- USB 2.0 port on the move (OTG)
- Connect standard peripherals to USB using any of the following:
    - One mini-A to standard A cable adapter
    - DVI-D using an HDMI to DVI-D adapter
    - MMC / SD / SDIO connector allows for a complete desktop experience.

Intel

Intel has designed two models of boards, the Galileo® and the Édison® for free hardware and software projects, but currently it has been discontinued in its development, there are other kits, but due to the costs they are not applied. Below, more technical information of these boards is given.

Intel Galileo

The second-generation Intel Galileo board provides a unique board controller for the maker community, students and professional developers. Based on the Intel Quark SoC X1000, a 32-bit Intel Pentium processor class system on a chip (SoC), the original Intel processor and native input/output capabilities of The Intel Galileo (Gen 2) board offers full function for a wide range of Arduino-certified applications and designed to be hardware and software pin-compatible with a wide range of Arduino Uno R3 shields.

The Intel Galileo Gen 2 board also offers a simpler and more cost-effective development environment compared to Intel Atom processor and Intel Core processor-based designs.

They use the Arduino software development environment to create programs for Galileo called "sketches."

Intel® Edison

The Intel Edison development platform is designed to reduce barriers to entry for a range of inventors, entrepreneurs, and consumer product designers to rapidly prototype and produce Internet of Things (IoT) and wearable computing products.

The Intel Edison compute module is designed for use with custom printed circuit boards.

Arduino

First of all, it is important to define that when we talk about Arduino. We are talking about three things:

- A free hardware board that incorporates a reprogrammable microcontroller and a series of socket pins (which are internally linked to the input/output pins of the microcontroller) that allow different sensors and actuators to be connected there in a very simple and convenient way. (Torrente Artero, 2013, p. 65).
- A software (more specifically, a "development environment") free, complimentary, and multiplatform (since it works on Linux, MacOS, and Windows) that we must install on our computer and that allows us to write, verify, and save ("load") in the memory of the microcontroller of the Arduino board the set of instructions that we want it to start executing. That is to say, it allows us to program it.

The standard way to connect our computer with the Arduino board to send and record these instructions is through a simple USB cable because most Arduino boards incorporate a connector of this type. (Torrente Artero, 2013, p. 66).

- A free programming language, "programming language" means any artificial language designed to express instructions (following certain syntactic rules) that can be carried out by machines.

Specifically, within the Arduino language, we find elements similar to many other existing programming languages (such as conditional blocks, repetitive blocks, variables, etc.), as well as different commands - also called "commands" or "functions" - that allow us to specify in a consistent and error-free way the exact instructions that we want to program in the microcontroller of the board. We write these commands using the Arduino development environment. (Torrente Artero, 2013, p. 66)

There is a wide variety of Arduino boards, but the one taken for this research is the Arduino UNO R3, which uses the ATmega328 microcontroller, in addition to all the features of the previous boards. The Arduino Uno uses the ATmega16U2 for USB handling instead of the 8U2; this allows faster transfer rates and more memory. No drivers are needed for Linux or Mac.

SDA and SCL pins are added near the AREF; moreover, there are two new pins near the RESET pin, one is the IOREF, which allows the shields to adapt to the voltage provided by the board, the other pin is not connected and is reserved for future purposes. The board works with all existing shields and will be able to adapt to new shields using these additional pins.

The open-source IDE can be downloaded free of charge and is currently available for Mac OS X, Windows, and Linux.

The characteristics of the plate are:

- ATmega328 microcontroller.

- Input voltage 7-12V.
- 14 digital I/O pins (6 PWM outputs).
- 6 analog inputs.
- 32k Flash memory.
- 16MHz clock speed.

Technical specification of the hardware used

The focus of the project is oriented towards the Arduino UNO so the rest of the hardware or modules were selected based on this development board.

Previously we detailed the characteristics of the Arduino board, for this reason we will now provide specific information about the module that is integrated to the motherboard to achieve the functionality required for this project, it is the eMGing SIM808 shield.

eMGing SIM808 Shield

The eMGing SIM808 shield is a very useful board for this project because it allows GSM, GPS, and GPRS connectivity by integrating multiple modules on a single shield, making it very easy to implement on our Arduino UNO motherboard.

Results

As for the results, several tests were carried out on the device; as a first step, a 12-volt, 2 Amp power source was connected to the Arduino UNO either with a charger or with a portable battery; it must be ensured that it turns on correctly and starts sending data to the web platform.

Once it has been verified that the device has been turned on correctly, it is enabled to perform the pertinent tests in a simple route in the field and observe how the route we are traveling is drawn on the map in real time.

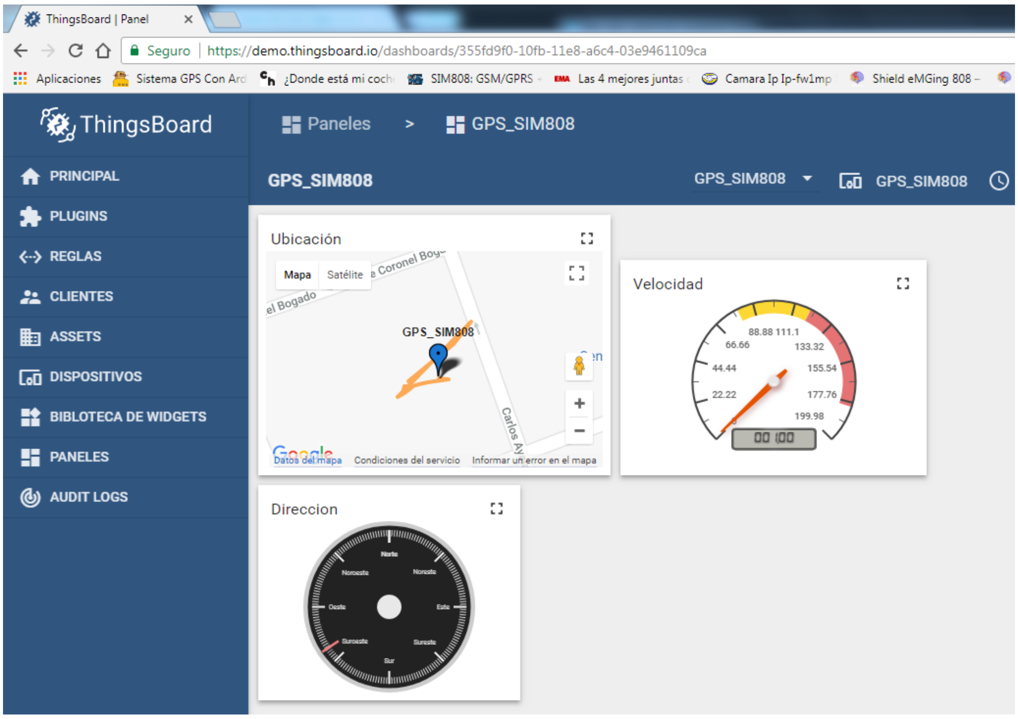


Figure 2. Simulation.
Note: Source: Own elaboration.

As can be seen in the illustration above, the map exceeded all expectations and was able to collect really accurate data on the trails traveled; in addition, the speed and direction data were considerably acceptable, thus fulfilling the objectives of this project.

In general, the device can be installed on any type of animal to be monitored and the results can be observed on the ThingsBoard web platform, thus obtaining precise control over them.

In summary, the GPS tracker was programmed to send data directly to ThingsBoard via HTTP POST requests, and we managed the data in a dashboard, where multiple devices and panels can be added, each with multiple widgets that provide a very good visual impression and have many customization options.

ThingsBoard has proven to be a very powerful tool for observing IoT data, it has a very simple interface, as is its configuration, allowing a simple and fast connection for devices.

Conclusions

The general objective of this research consisted in the development of a system capable of requesting and receiving positioning coordinates from satellites dedicated to global geo-positioning and being able to send the information obtained to a WEB platform to be able to visualize that data there, to track the animal, and try to prevent its theft. This was carried out through the implementation of free programming hardware and software, the elements used were the Arduino UNO, the SIM808 shield, the Arduino development environment, and the ThingsBoard internet of things platform.

The Arduino UNO R3 is an open-source physical computing platform based on a simple input/output board that has its own development IDE, which can be integrated with different modules to perform an infinite number of functions and actions.

We used, together with the Arduino UNO, the SIM808 shield developed by an Argentine company, which has integrated on its board the GSM, GPRS, and GPS modules necessary to meet the objectives proposed for this research. This board is designed to be perfectly embedded on the Arduino board, thus facilitating its assembly.

The ThingsBoard web platform is dedicated to the Internet of Things, meaning that any device that sends information through the Internet can connect to it; its main objective is to be able to manage and visualize the information they send.

The implementation of free hardware dedicated to the field of geo positioning allows the creation of new services over the network, focused on tracking and allowing its geo reference in real time, as well as the respective sending of information through the mobile network, which allows IoT platforms a real-time reading of the coordinates received and to translate these data into different widgets.

The research carried out was very satisfactory because it led to know several free hardware options that could be used in different ways and with different characteristics, but that would lead to a similar result.

To conclude, the research fulfilled the objectives proposed at the beginning, and the data obtained were well accepted by the herdswomen.

Bibliographic references

Anderson, D. J., Carmichael, A., (2016). *Kanban Esencial Condensado*. (5ta Ed.). Washington: Lean Kanban University Press.

Albanés, A. J. (21 de 05 de 2011). *Evolución en el Desarrollo Web*. http://www.Isi.us.es/docencia

Barsky, O. (1992), Políticas agrícolas en la Argentina en el contexto del ajuste. Ruralia, 3, 7-34.

Barsky, O.; Gelman, J. (2002). *Historia del agro argentino. Desde la conquista hasta fines del siglo XX*. Grijalbo-Mondadori. Argentina.

Basualdo, E. & N. Arceo (2006). Evolución y situación actual del ciclo ganadero en la Argentina. *Revista Realidad Económica, 221*.

Ghio M. Gina. (2008). Sistema de Posicionamiento Global (GPS): Sociedad de Especialistas Latinoamericanos. *Percepción Remota y Sistemas de Información Espacial Sesión, 94*.

Held, Gilbert. (1997). *Diccionario de tecnología de las comunicaciones*. Paraninfo.

Hernández Sampieri R., Fernández Collado C., Lucio P., (2010). *Metodología de la Investigación*. McGraw-Hill.

Lattuada M. (1997). *El sector agropecuario Argentino hacia fines del milenio.* Transición e incertidumbre.

Pallás Areny, Ramón. (2004). *Sensores y acondicionadores de señal*. Maracombo.

Passaniti,M. V. (2011). *Estudio del sector de ganado y carne vacunos argentino y políticas públicas*. [Trabajo Final]. Universidad Católica Argentina.

Pressman, R. (2002). *Ingeniería de Software. Un enfoque práctico*. Mc. Graw Hill.

Rey, Eugenio. (1995). *Telecomunicaciones móviles*. Alfaomega.

Sierra Pérez, M., Galocha Iragüen, B., Fernández Jambrina, J. l., & Sierra Castañer, M. (2003). *Electrónica de Comunicaciones*. Pearson Educación, S. A.

Sommerville, I. (2011). *Ingeniería de Software*. Pearson.

Receipt date: 09/29/2021
Revision date: 12/12/2021
Acceptance date: 02/27/2022

Annexes

Annex 1

Assembly

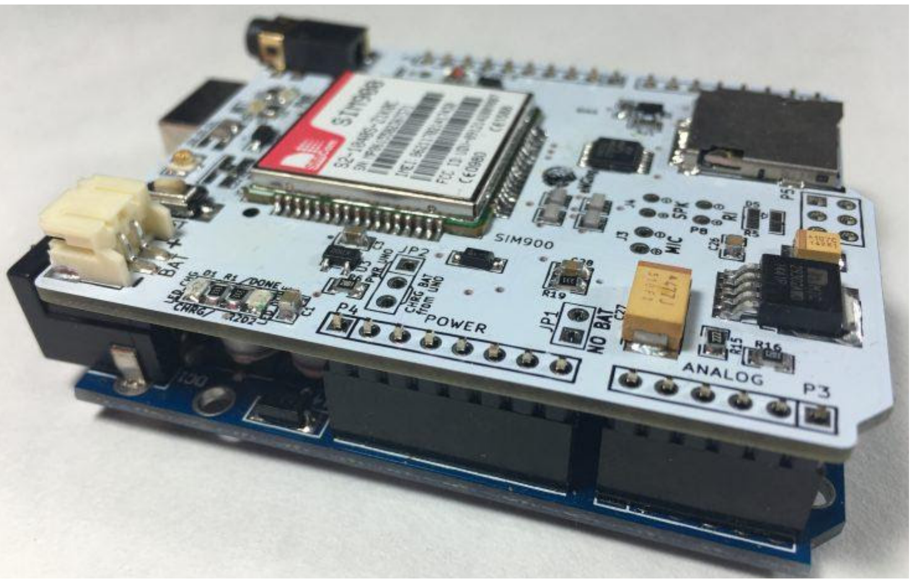


Figure 3. Assembly step 1.
Note: Source: Own elaboration.

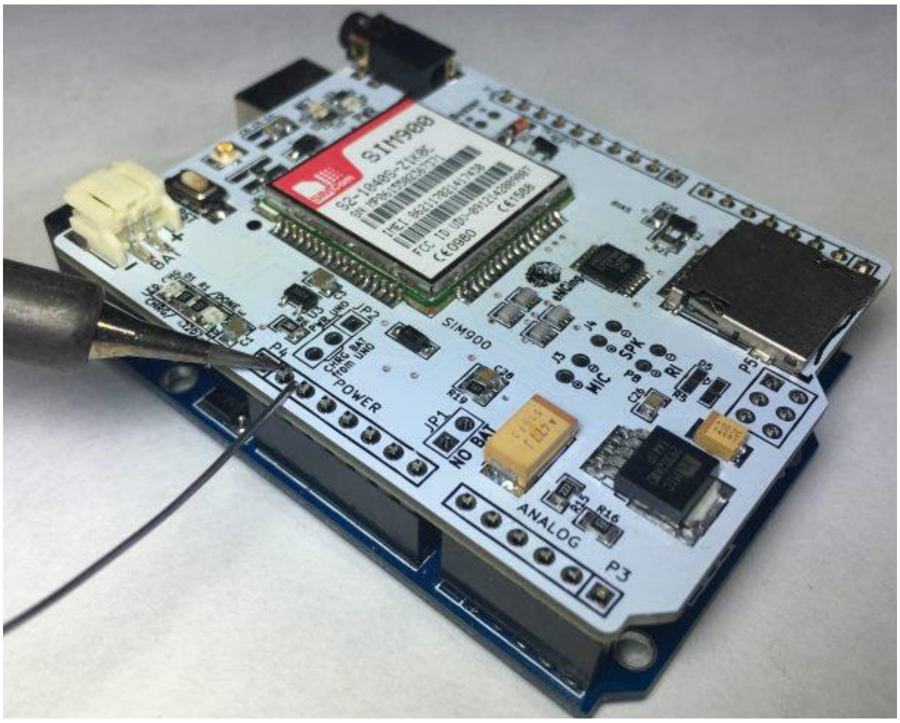


Figure 4. Assembly step 2.
Note: Source: Own elaboration.

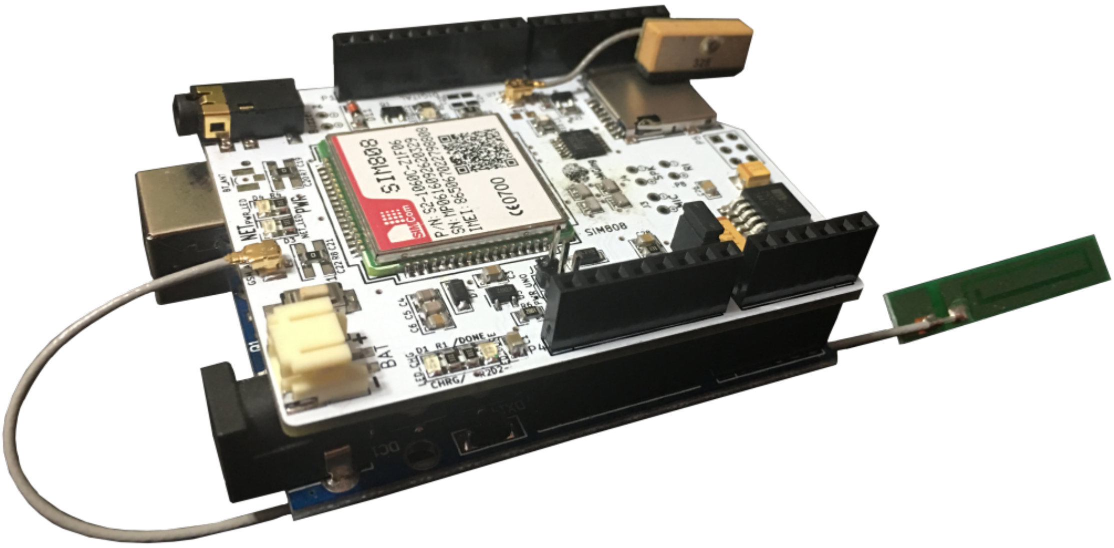
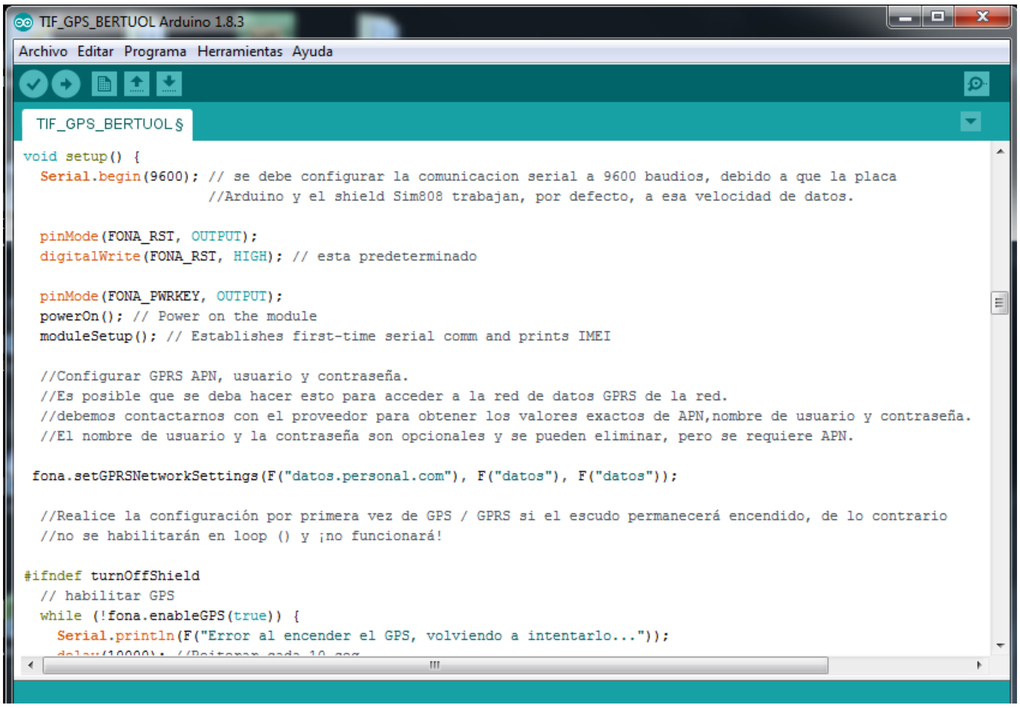


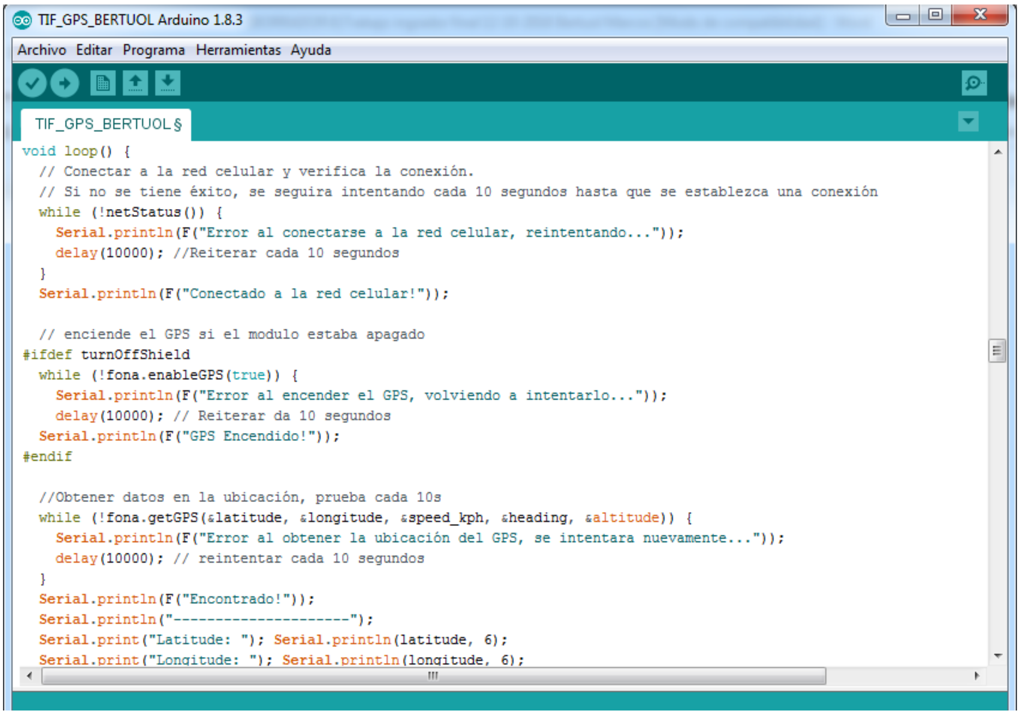
Figure 5. *Assembly step 3.*
Note: Source: Own elaboration.

Annex 2
GPS Software Development on Arduino



```
void setup() {
  Serial.begin(9600); // se debe configurar la comunicacion serial a 9600 baudios, debido a que la placa
                      //Arduino y el shield Sim808 trabajan, por defecto, a esa velocidad de datos.

  pinMode(FONA_RST, OUTPUT);
  digitalWrite(FONA_RST, HIGH); // esta predeterminado

  pinMode(FONA_PWRKEY, OUTPUT);
  powerOn(); // Power on the module
  moduleSetup(); // Establishes first-time serial comm and prints IMEI

  //Configurar GPRS APN, usuario y contraseña.
  //Es posible que se deba hacer esto para acceder a la red de datos GPRS de la red.
  //debemos contactarnos con el proveedor para obtener los valores exactos de APN,nombre de usuario y contraseña.
  //El nombre de usuario y la contraseña son opcionales y se pueden eliminar, pero se requiere APN.

 fona.setGPRSNetworkSettings(F("datos.personal.com"), F("datos"), F("datos"));

  //Realice la configuración por primera vez de GPS / GPRS si el escudo permanecerá encendido, de lo contrario
  //no se habilitarán en loop () y ¡no funcionará!

#ifndef turnOffShield
  // habilitar GPS
  while (!fona.enableGPS(true)) {
    Serial.println(F("Error al encender el GPS, volviendo a intentarlo..."));
```

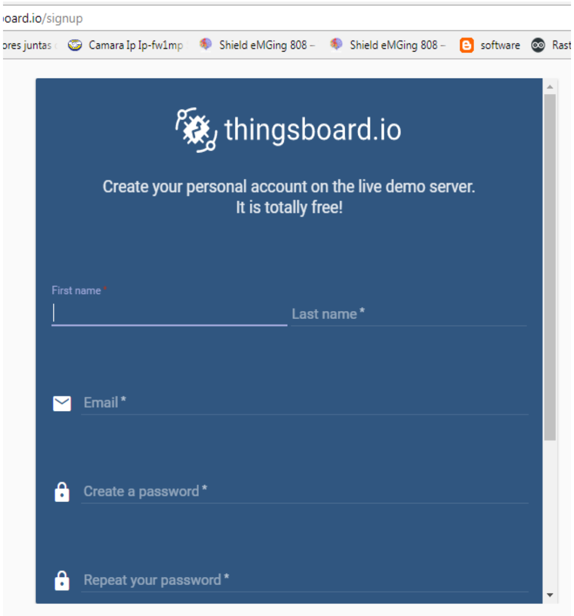
Figure 6. Setup subroutine.
Note: Source: Own elaboration.



```
void loop() {
  // Conectar a la red celular y verifica la conexión.
  // Si no se tiene éxito, se seguira intentando cada 10 segundos hasta que se establezca una conexión
  while (!netStatus()) {
    Serial.println(F("Error al conectarse a la red celular, reintentando..."));
    delay(10000); //Reiterar cada 10 segundos
  }
  Serial.println(F("Conectado a la red celular!"));

  // enciende el GPS si el modulo estaba apagado
#ifdef turnOffShield
  while (!fona.enableGPS(true)) {
    Serial.println(F("Error al encender el GPS, volviendo a intentarlo..."));
    delay(10000); // Reiterar da 10 segundos
  Serial.println(F("GPS Encendido!"));
#endif

  //Obtener datos en la ubicación, prueba cada 10s
  while (!fona.getGPS(&latitude, &longitude, &speed_kph, &heading, &altitude)) {
    Serial.println(F("Error al obtener la ubicación del GPS, se intentara nuevamente..."));
    delay(10000); // reintentar cada 10 segundos
  }
  Serial.println(F("Encontrado!"));
  Serial.println("---------------------");
  Serial.print("Latitude: "); Serial.println(latitude, 6);
  Serial.print("Longitude: "); Serial.println(longitude, 6);
```

Figure 7. Loop subroutine.
Note: Source: Own elaboration.

Annex 3
Configuration of the IoT Platform, ThingsBoard



Create your personal account on the live demo server.
It is totally free!

Figure 7. IoT platform configuration, step 1.

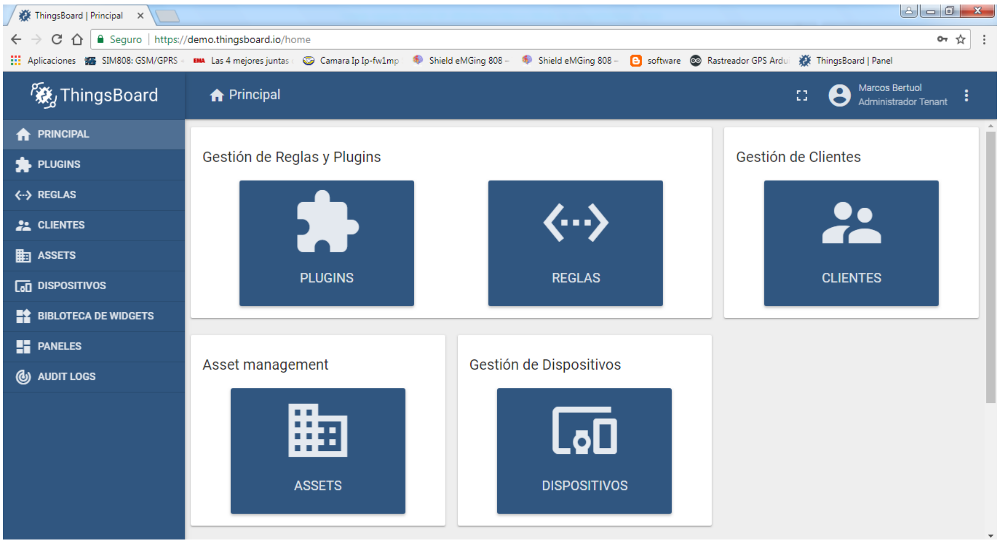


Figure 8. IoT platform configuration, step 2.
Note: The "Devices" tab on the left side was selected.

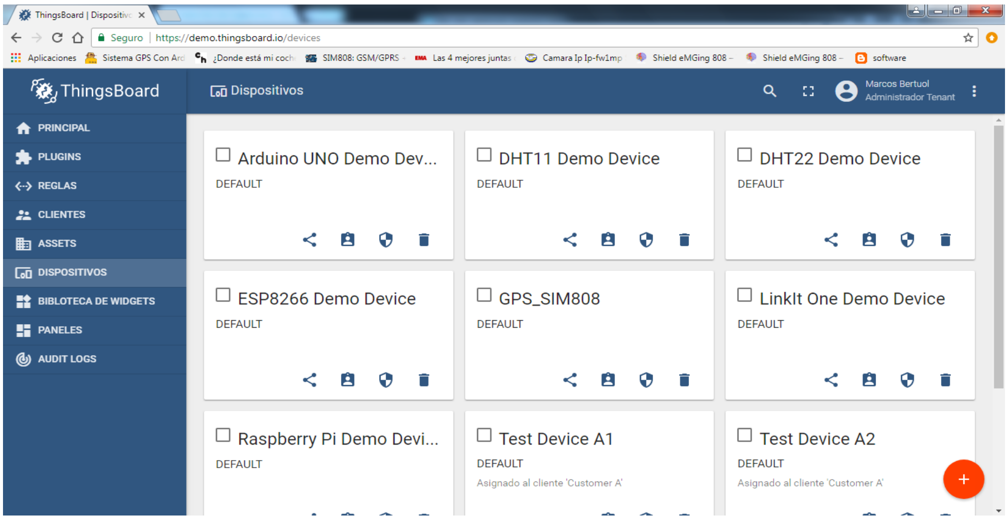


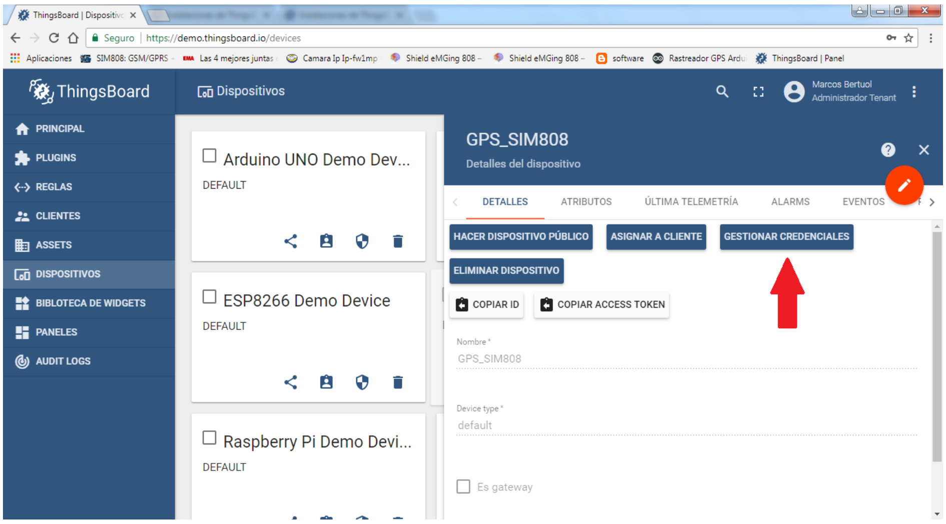
Figure 9. IoT platform configuration, step 3.
Note: Source: Own elaboration.

Nombre *
GPS_SIM808

Device type *
default

☐ Es gateway

Figure 10. IoT platform configuration, step 4.
Note: Source: Own elaboration.

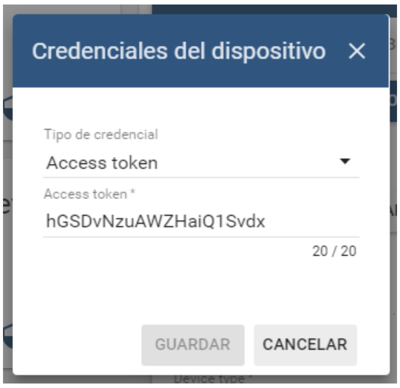


Nombre *
GPS_SIM808

Device type *
default

☐ Es gateway

Figure 11. IoT platform configuration, step 5.
Note: Source: Own elaboration.



Tipo de credencial
Access token

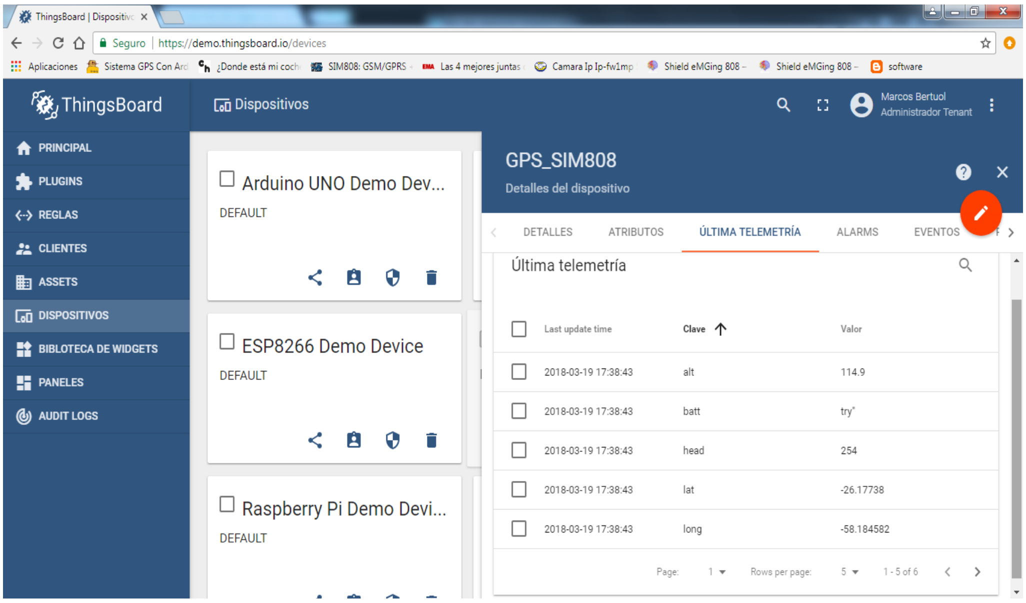
Access token *
hGSDvNzuAWZHaiQ1Svdx

20 / 20

Figure 12. IoT platform configuration, step 6.
Note: Source: Own elaboration.

Annex 4
Verification of data reception

It is entered in the same device details page, in the "Last telemetry" tab.



Última telemetría

☐	Last update time	Clave ↑	Valor
☐	2018-03-19 17:38:43	alt	114.9
☐	2018-03-19 17:38:43	batt	try"
☐	2018-03-19 17:38:43	head	254
☐	2018-03-19 17:38:43	lat	-26.17738
☐	2018-03-19 17:38:43	long	-58.184582

Figure 13. Verification of data reception.
Note: Source: Own elaboration.

Annex 5

Board configuration

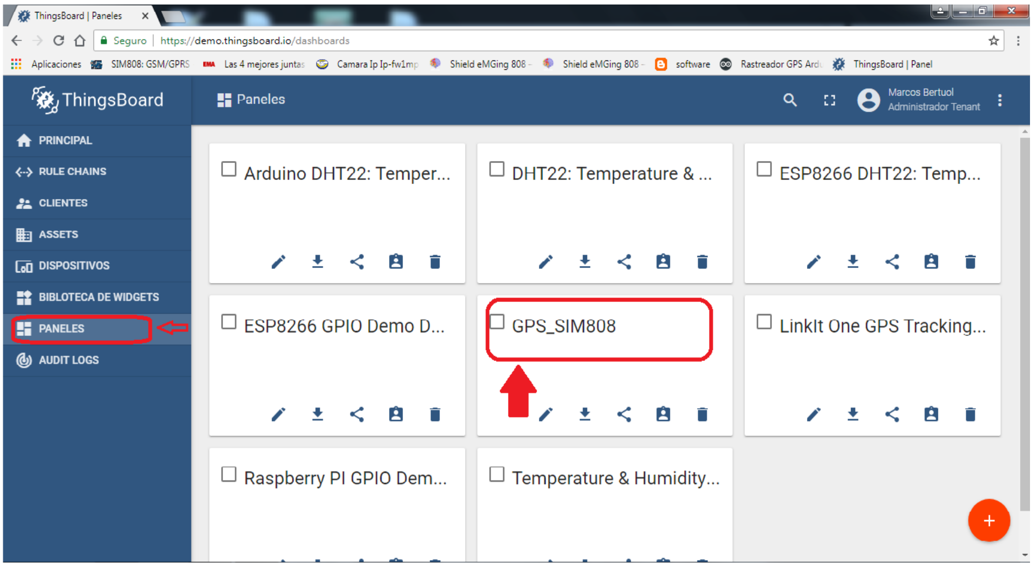


Figure 14. Board configuration, step 1.
Note: Source: Own elaboration.

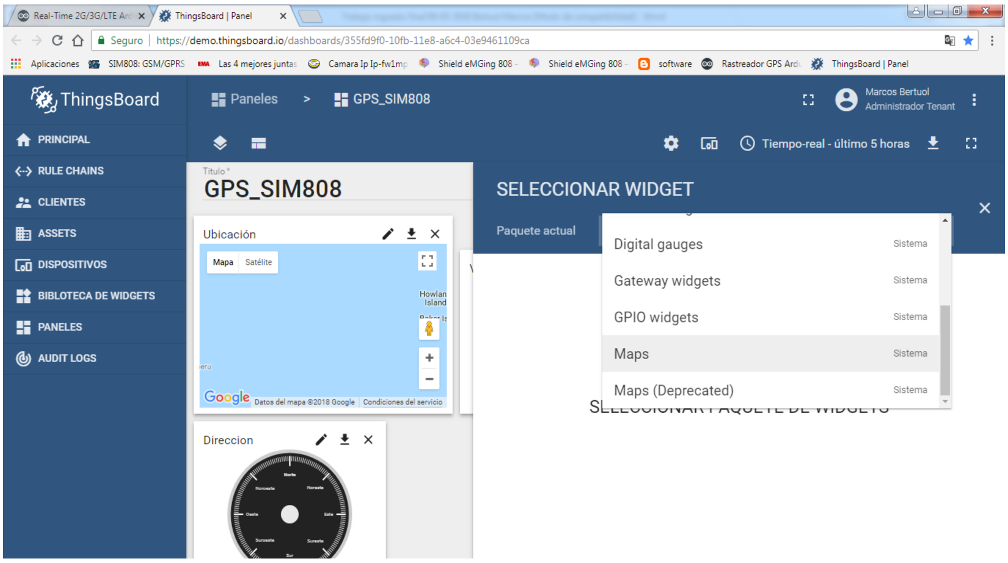
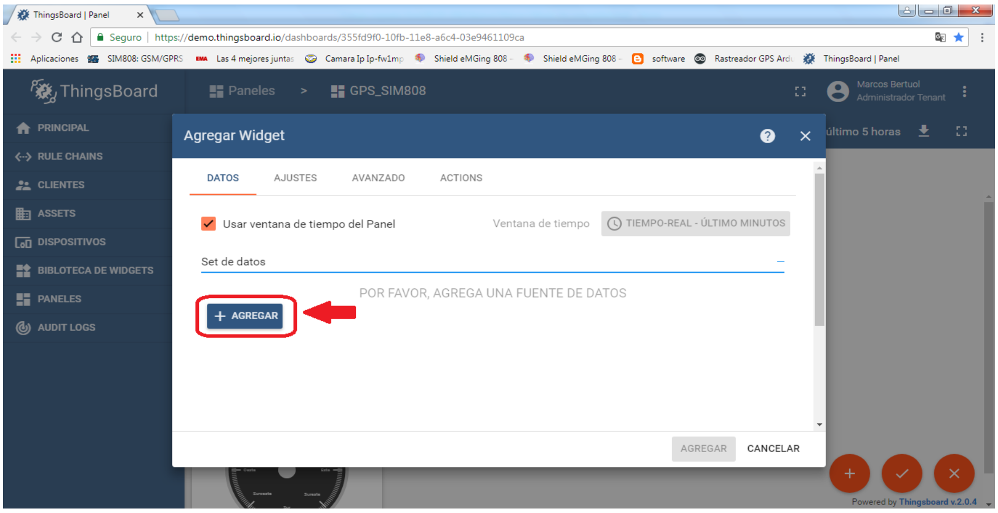


Figure 15. Board configuration, step 2.
Note: Source: Own elaboration. Selecting this loaded the previews for all the different types of maps that can be chosen.



☑ Usar ventana de tiempo del Panel

Ventana de tiempo: TIEMPO-REAL - ÚLTIMO MINUTOS

Set de datos

POR FAVOR, AGREGA UNA FUENTE DE DATOS

Figure 16. Board configuration step 3.

Note: Source: Own elaboration. Once the widget has been selected and configured, the prototype can be used.