

CARACTERIZACIÓN DE LA VARIABLE TEMPERATURA CORPORAL PARA
DETERMINAR EL ESTADO DE ESTRO EN GANADO BOVINO TÍPICO DEL
DEPARTAMENTO DEL CAUCA MEDIANTE TECNOLOGÍA IOT.



INTEGRANTES:
MARIA DE LOS ANGELES ROSERO MONTILLA.
JUAN MANUEL VIVAS TORRES.

DIRECTOR
MSc. JUAN PABLO DIAGO RODRIGUEZ

CODIRECTOR
MSc. YAMIR HERNANDO BOLAÑOS MUÑOZ

CORPORACION UNIVERSITARIA AUTONOMA DEL CAUCA.
POPAYÁN - CAUCA.

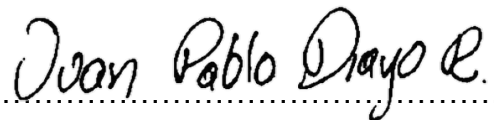
2021



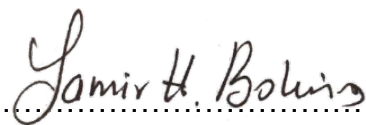
.....
Yesid Enrique Castro Caicedo
Jurado

Daniel R. Ortega

.....
Daniel Rodrigo Ortega Alegría
Jurado



.....
MSc. Juan Pablo Diago Rodríguez
Director de trabajo de grado



.....
MSc. Yamir Hernando Bolaños Muños
Codirector de trabajo de grado

CONTENIDO

1. INTRODUCCIÓN	10
1.1. PLANTEAMIENTO DEL PROBLEMA	10
1.2. OBJETIVOS	11
1.2.1 OBJETIVO GENERAL	11
1.2.2 OBJETIVOS ESPECÍFICOS	11
2. MARCO TEÓRICO	12
2.1 VARIABLES BIOLÓGICAS QUE PERMITEN DETECTAR EL ESTADO DE ESTRO EN GANADO BOVINO.	12
2.2 TEMPERATURA	12
2.3 POSICIÓN GEOGRÁFICA	13
2.4 FUNCIONAMIENTO DE LOS GPS	14
2.5 FUNCIONAMIENTO DE LOS PROTOCOLOS DE COMUNICACIÓN GPS	15
2.6 DECODIFICACIÓN DE LOS PROTOCOLOS GPS	15
2.6.1 TRAMA \$GNRMC	15
2.6.2 TRAMA \$GNGGA	16
2.7 TECNOLOGÍA DE COMUNICACIÓN IOT	16
2.7.1 SIGFOX	16
2.7.2 LORAWAN	17
2.7.3 ZIGBEE	18
2.7.4 WIFI	19
3. DISEÑO DEL SISTEMA ELECTRÓNICO	21
3.1 DISEÑO HARDWARE DEL SISTEMA	21
3.1.1 DEFINICIÓN DE REQUERIMIENTOS	21
3.1.2 DISEÑO ELECTRÓNICO	22
3.1.3 SELECCIÓN DE COMPONENTES	23
3.1.3.1 SENSORES DE TEMPERATURA	23
3.1.3.2 SENSORES DE GPS	25
3.1.3.3 MICROCONTROLADOR	26
3.1.3.4 SELECCIÓN DE SD	27
3.1.3.5 SELECCIÓN FUENTE DE ALIMENTACION	28
3.1.3.6 SELECCIÓN TECNOLOGÍA DE COMUNICACIÓN PARA IOT	31
3.1.3.6.1 SELECCIÓN DEL TRANSECTOR DE COMUNICACIÓN SIGFOX	33
3.1.4 DISEÑO DEL PLANO ESQUEMÁTICO	34
3.1.5 ESTRATEGIA ELECTRONICA DE CONTROL PARA BAJO CONSUMO	37

3.1.6 DISEÑO MECANICO	39
3.2 DISEÑO SOFTWARE DEL SISTEMA	41
3.2.1 DEFINICIÓN DE REQUERIMIENTOS FUNCIONALES, CASOS DE USO	41
3.2.1.1 DIAGRAMA CASOS DE USO	42
3.2.2 SELECCIÓN DE HERRAMIENTAS DE DESARROLLO	43
3.2.2.1 SELECCIÓN DEL ENTORNO DE DESARROLLO Y PROGRAMACIÓN	43
3.2.2.2 SELECCIÓN PLATAFORMA IOT	44
3.2.3 IMPLEMENTACIÓN DE HERRAMIENTAS SOFTWARE	46
3.2.3.1 DESARROLLO DE FIRMWARE CON LA HERRAMIENTA ARDUINO IDE	46
3.2.3.2 MANEJO DEL BACKEND DE SIGFOX	49
3.2.3.3 MANEJO DE LA PLATAFORMA UBIDOTS	49
3.3 IMPLEMENTACIÓN DE SISTEMA ELECTRÓNICO	51
3.3.1 PROTOTIPADO ELECTRÓNICO – PRUEBAS PARCIALES	52
3.3.2 PRUEBAS DE LABORATORIO DE SISTEMA ELECTRÓNICO – CIRCUITO INTEGRADO	57
4. EVALUACIÓN DEL SISTEMA ELECTRÓNICO	58
4.1 DISEÑO DE PROTOCOLO DE PRUEBAS	58
4.2 RESULTADOS OBTENIDOS DEL SISTEMA ELECTRÓNICO	59
5. ANÁLISIS DE DATOS	63
5.1 ADQUISICIÓN Y TRANSFORMACIÓN DE LOS DATOS	63
5.2 SEGMENTACIÓN	63
5.3 PRUEBAS ESTADÍSTICAS	66
5.4 RESULTADOS OBTENIDOS ANÁLISIS DE DATOS	68
5.5 PLANTEAMIENTO DEL ALGORITMO PARA ESTIMAR EL ESTADO DE CELO	71
6. CONCLUSIONES	73
7. RECOMENDACIONES Y TRABAJOS FUTUROS	74
8. BIBLIOGRAFÍA	76

LISTA DE ABREVIATURAS

GPS	Global Positioning System
ICA	Instituto Colombiano Agropecuario
IoT	Internet of Things
LoRaWAN	Low Power Wide Area Network
PIB	Producto Interno Bruto
NMEA	National Marine Electronics Association.
ASCII	American Standard Code For Information Interchange
RTC	Reloj en Tiempo Real
PCB	Placa de Circuito Impreso
UML	Lenguaje de Modelado Unificado

LISTA DE FIGURAS

Ilustración 1. Signos de estro en bovinos.....	12
Ilustración 2. Estructura red Sigfox..	17
Ilustración 3. Estructura red LoRaWAN.....	18
Ilustración 4. Esquema red Zigbee.....	19
Ilustración 5. Estructura red Wifi..	20
Ilustración 6. Plataforma de monitoreo IoT.....	23
Ilustración 7. Microcontrolador ATSAMD21G18A..	27
Ilustración 8. Comunicación SPI..	28
Ilustración 9. Batería LiPo 850 mAh.....	31
Ilustración 10. Mapa de cobertura Sigfox.....	32
Ilustración 11. Diagrama en bloques del circuito electrónico.....	35
Ilustración 12. Diseño Electrónico en Eagle (SCH)..	36
Ilustración 13. Diseño electrónico en Eagle (BRD).....	36
Ilustración 14. PCB Ensamblado.....	37
Ilustración 15. Diseño 3D Solid Edge.....	40
Ilustración 16. Diseño 3D PovRay.....	40
Ilustración 17. Diseño mecánico del sistema.....	41
Ilustración 18. Diagrama casos de uso UML.....	43
Ilustración 19. Diagrama de implementación software.....	46
Ilustración 20. Estructura de mensaje Sigfox..	46
Ilustración 21. Formato de punto flotante IEEE-754.....	47
Ilustración 22. Tramas de envío.....	47
Ilustración 23. Backend de Sigfox.....	49
Ilustración 24. Dashboard Ubidots.....	50
Ilustración 25. Geocerca Ubidots.....	50
Ilustración 26. Envío de datos con ESP32.....	51
Ilustración 27. Placa pcb con módulo Wisol y ATSAMD21.....	51
Ilustración 28. Placa pcb con módulo Wisol, ATSAMD21 y L80-R.....	52
Ilustración 29. Sensor de temperatura MLX90614.....	52
Ilustración 30. Diagrama de flujo MLX90614.....	53
Ilustración 31. Prueba medición temperatura MLX90614 y CCIR1001.....	54
Ilustración 32. GPS L80-R.....	54
Ilustración 33. Diagrama de flujo L80-R.....	55
Ilustración 34. Prueba medición de la ubicación entre L80-R Y Google Maps.....	56
Ilustración 35. Tablero principal Ubidots.....	57
Ilustración 36. Primera versión PCB ensamblado.....	59
Ilustración 37. Segunda versión PCB ensamblado.....	60
Ilustración 38. Primera versión collar.....	60
Ilustración 39. Problemas primera versión collar.....	61
Ilustración 40. Ajustes collar.....	61
Ilustración 41. Grafica de datos 11/09/21.....	62
Ilustración 42. Grafica Temperaturas 8/09/21.....	62

Ilustración 43. Datos almacenados y transformados en archivo “.Csv”..	63
Ilustración 44. Datos segmentados para el análisis.....	64
Ilustración 45. Temperaturas obtenidas del 13 y 19 de septiembre.....	64
Ilustración 46. Valor absoluto de las temperaturas del 13 y 19 de septiembre.	65
Ilustración 47. Aceleraciones del 13 y 19 de septiembre.....	66
Ilustración 48. Mapa conceptual – Análisis estadístico.....	66
Ilustración 49. Series de temperatura para el análisis..	67
Ilustración 50. Boxplot de las temperaturas del 13 y 19 de septiembre.	67
Ilustración 51. Test de Kolmogorov - Smirnov.....	68
Ilustración 52. Test de Wilcoxon.	68
Ilustración 53. Resultados Test de Kolmogorov de los días 13 y 19 septiembre.	69
Ilustración 54. Resultados Test de Wilcoxon de los días 13 y 19 septiembre.....	70
Ilustración 55. Resultados Test de Wilcoxon de los días 13 y 19 septiembre con los parámetros “Left” y “Right”.....	70
Ilustración 56. Resultados del test de Kolmogorov y Wilcoxon para el valor absoluto.	71
Ilustración 57. Resultados del test de Kolmogorov y Wilcoxon para la aceleración.	71
Ilustración 58. Algoritmo estimación de Celo.....	72
Ilustración 59. Configuración Backend Sigfox - Device Type.	82
Ilustración 60. Configuración Backend Sigfox - Device Type Name.....	82
Ilustración 61. Configuración Backend Sigfox - Device Type Edit.	83
Ilustración 62. Configuración Backend Sigfox - Device Type Payload Display..	83
Ilustración 63. Configuración Backend Sigfox - Device Type CallBack ..	84
Ilustración 64. Configuración Backend Sigfox - Device Type Callback Name.	84
Ilustración 65. Configuración Backend Sigfox - Callback Seleccion.....	84
Ilustración 66. Configuración Backend Sigfox - Callback Nuevo.	85
Ilustración 67. Configuración Backend Sigfox - Custom Callback.	85
Ilustración 68. Configuración Backend Sigfox - Configuración Callback.....	85
Ilustración 69. Configuración Backend Sigfox - Callback Fin.....	86
Ilustración 70. Configuración Ubidots - Web..	86
Ilustración 71. Configuración Ubidots - Tipo de cuenta.	87
Ilustración 72. Configuración Ubidots - Registro.	88
Ilustración 73. Configuración Ubidots - Devices.	88
Ilustración 74. Configuración Ubidots - Functions.	89
Ilustración 75. Configuración Ubidots - Nueva Functions.	89
Ilustración 76. Configuración Ubidots - Configuración parámetros functions.....	90
Ilustración 77. Configuración Ubidots - HTTPS Functions.....	90
Ilustración 78. Configuración Ubidots - Programación Functions..	91
Ilustración 79. Configuración Ubidots - Api.	91
Ilustración 80. Comunicación Sigfox - Ubidots.	92
Ilustración 81. Configuración Ubidots - Dashboard.	93
Ilustración 82. Configuración Ubidots - Nuevo Dashboard.	93
Ilustración 83. Configuración Ubidots - Configuración Dashboard.....	94
Ilustración 84. Configuración Ubidots - Widgets Dashboard.....	94

Ilustración 85. Configuración Ubidots - Ajuste Widget Dashboard.....	95
Ilustración 86. Configuración Ubidots - Visualización Variables.	95
Ilustración 87. Finca elegida para realizar las pruebas de campo.	96
Ilustración 88. Selección del bovino para realizar las pruebas..	97
Ilustración 89. Signos fisiológicos de celo - 19 Septiembre.	98
Ilustración 90. Signos Fisiológicos del celo - 19 de Septiembre.	98

LISTA DE TABLAS

Tabla 1. Temperaturas típicas del ganado.....	13
Tabla 2. Protocolos GPS.....	14
Tabla 3. Requerimientos electrónicos del sistema.....	21
Tabla 4. Sensores de temperatura.....	24
Tabla 5. Sensores de GPS..	25
Tabla 6. Microcontroladores.....	26
Tabla 7. Clasificación tarjeta SD	28
Tabla 8. Características de la batería.....	29
Tabla 9. Consumo de corriente del sistema.....	29
Tabla 10. Variables para calcular el consumo energético..	30
Tabla 11. Tecnologías asociadas a IoT.....	31
Tabla 12. Transceptores..	33
Tabla 13. Consumo eléctrico del sistema - Estrategia bajo consumo.....	38
Tabla 14. Variables estrategia electrónica de control..	38
Tabla 15. Requerimientos funcionales.	42
Tabla 16. Comparación de plataformas IoT..	44
Tabla 17. Tramas Sigfox..	48
Tabla 18. Comandos AT-Modulo Wisol.....	48
Tabla 19. Comparación de medición de temperatura entre CCIR1001 Y LMX90614..	53
Tabla 20. Comparación de medición de ubicación GPS, entre L80-R Y Google Maps.....	56

1. INTRODUCCIÓN

El propósito de este trabajo es estimar el estado de estro en ganado bovino, a través del monitoreo y caracterización de la variable temperatura. Esto se realizará con el apoyo de un sistema electrónico basado en el concepto de internet de las cosas, que permitirá la medición de diferentes variables para su posterior análisis.

Para el diseño del sistema se requiere diferentes componentes tecnológicos como la conectividad a una red IoT (Internet Of Things), la cual permite la transmisión y recepción de datos a una plataforma web; además de contar con un sensor de temperatura para la medición corporal del animal; también se adicionó un módulo GPS (Global Positioning System) que permite la recolección de datos relevantes en cuanto a la ubicación del bovino y un dispositivo de control para administrar todos los procesos lógicos que requiere el sistema.

Por lo tanto para caracterizar la variable temperatura, es necesario hacer uso de un método de análisis de datos, que permita procesar e interpretar la información recogida por el dispositivo electrónico. Finalmente la información medida por los sensores será cargada a una plataforma digital, desde la cual el usuario podrá observar y tomar las decisiones sobre el estado del bovino (hembra).

1.1. PLANTEAMIENTO DEL PROBLEMA

La ganadería en Colombia es una actividad de vital importancia pues de ella depende una parte de la economía del país, con la elaboración de productos importantes de primera necesidad y de la canasta familiar como los productos lácteos y cárnicos, además, se tiene otros productos igualmente importantes como los cueros. En este sentido la actividad ganadera sobresale en el campo colombiano con relación a otros sectores como el café, las flores, los porcinos, entre otros, representando el 1.6% del PIB a nivel nacional [1] ubicando así al país en la posición 11 de producción de carne y en la posición 12 de producción de leche en el mundo [2]; por tal razón, es importante el cuidado que se debe tener de este sector, resaltando que los ganaderos en Colombia tienen sus principales preocupaciones en aspectos como (salud, alimentación y reproducción). Conllevando para ellos un gran esfuerzo humano y a su vez económico.

A nivel nacional la ganadería está constituida aproximadamente por 28.245.262 bovinos según el ICA (Instituto Colombiano Agropecuario) [3], ubicados en su mayoría en los departamentos de Antioquia, Caquetá, Meta y Casanare, como los más destacados, así como también, existen regiones donde esta práctica es baja pero no menos importante, como es el caso del departamento del Cauca en el cual se tienen alrededor de 316.000 cabezas de ganado repartido en 26.000 predios del territorio del departamento [4], el mayor número de animales se encuentran concentrados en los municipios de Patía, Puracé, Popayán, Santander de Quilichao, Mercaderes,

Sotar, Silvia y Totor, representando as aproximadamente un 50% de la totalidad de animales en la regin que constituye el departamento [5].

El departamento del Cauca cuenta con condiciones geogrficas e hdricas propicias para el desarrollo de la ganadera [6]. Sin embargo, los ganaderos se enfrentan a diversos problemas con sus animales, uno de ellos es la identificacin del estro o estado de celo en el bovino, ya que puede ser complejo detectarlo, generalmente se hace uso de la experticia de los ganaderos quienes mediante una inspeccin visual y anlisis del comportamiento del animal pueden intuir si est en ese estado, en algunos casos se complementa con el uso de mtodos tecnolgicos como (termmetros, crayolas, bozal marcador, arns marcador) [7], facilitando su deteccin. Estos elementos entregan informacin de variables corporales como la temperatura, la cual tiende a cambiar cuando el bovino se encuentra en celo. Adems, existen otros de los problemas que enfrentan los ganaderos como la falta de conocimiento relacionado con el tema y el poco tiempo del que disponen para prestarle atencin a esta actividad, pues el estro dura muy poco tiempo alrededor de 10 a 26 horas aproximadamente [8], generalmente se presenta en altas horas de la noche [9], dificultando an ms la deteccin del celo, generando una alteracin en la produccin de leche y el proceso reproductivo, por consiguiente una posible prdida econmica.

Por lo anterior, se evidencia una necesidad imperiosa de tener un sistema que le permita a los ganaderos tener la informacin suficiente para poder detectar el celo en un bovino (Hembra), evitando afectaciones al desarrollo de su labor, segn lo anterior surge la siguiente pregunta de investigacin: Cmo determinar el comportamiento temporal en la variable temperatura corporal durante el ciclo frtil de bovinos en la regin del departamento del Cauca, utilizando instrumentacin electrnica apropiada?.

1.2. OBJETIVOS

1.2.1 OBJETIVO GENERAL

Caracterizar la variable temperatura corporal para determinar el estado de estro en ganado bovino tpico de la regin del Cauca mediante tecnologa IoT.

1.2.2 OBJETIVOS ESPECFICOS

- Desarrollar un sistema electrnico IOT para el monitoreo de la variable temperatura corporal del ganado bovino en el departamento del Cauca.
- Proponer un algoritmo para la deteccin del estado de estro mediante la variable temperatura corporal en un bovino tpico del departamento del Cauca.
- Evaluar el algoritmo propuesto mediante un procedimiento experimental en campo utilizando una muestra piloto.

2. MARCO TEÓRICO

En esta sección se mencionan aspectos importantes que permitirán contextualizar sobre las temáticas que soportaran el proyecto de investigación, describiendo los conceptos de variables sensadas como lo son la temperatura corporal del bovino, y la posición geográfica, así como también los protocolos y tecnologías usadas por las redes de comunicación IoT y todo lo relacionado con los sistemas GPS.

2.1 VARIABLES BIOLÓGICAS QUE PERMITEN DETECTAR EL ESTADO DE ESTRO EN GANADO BOVINO.

El ganado bovino (hembra), presenta diferentes cambios en su comportamiento y morfología cuando entra en un estado de estro, los cuales se ven reflejados primordialmente en su temperatura corporal (aumenta), su actividad física (la vaca se muestra inquieta), ritmo cardíaco (se acelera), niveles hormonales, ubicación, entre otras [10] ver Ilustración 1. Es importante mencionar que no todas estas variables son parametrizables, esto quiere decir que no es posible medirlas constantemente mediante el uso de sistemas electrónicos portables, como es el caso de los niveles hormonales, por ejemplo, por lo tanto, para esta investigación se tendrán en cuenta sólo dos de las variables cuantificables: temperatura y ubicación geográfica, estas se especificarán más en detalle a continuación [9].

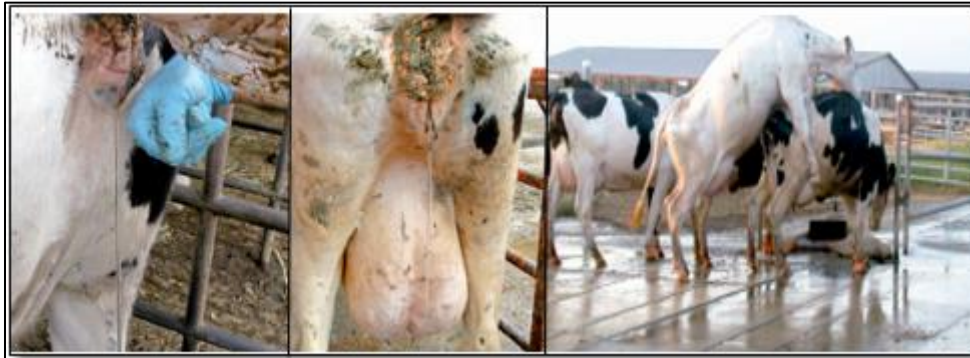


Ilustración 1. Signos de estro en bovinos. Fuente [10].

2.2 TEMPERATURA

En los animales de sangre caliente, la temperatura tiende a ser constante [11] sin embargo, al presentarse una alteración considerable en ella, se puede clasificar en estos dos estados:

Hipertermia: Sucede cuando la temperatura corporal se ha elevado sobre el nivel promedio considerado como normal del cuerpo del animal.

Hipotermia: Sucede cuando la temperatura corporal del animal se encuentra por debajo de los valores normales [11].

Como se puede observar en la Tabla 1, se presentan los rangos de temperatura entre los cuales varia para los animales desde valores mínimos, medios, y máximos, teniendo en cuenta si son adultos o jóvenes.

*Tabla 1.
Temperaturas rectales típicas del ganado. Fuente [12].*

ETAPA PRODUCTIVA	MÍNIMA	MEDIA	MÁXIMA
Vacunos adultos	37,7°C	38,5°C	39,0°C
Vacunos jóvenes	38,5°C	39,0°C	39,5°C

Algunos estudios han demostrado que la actividad física de las vacas y por ende su temperatura, cambian significativamente durante el estado de estro. Por lo que su medición debe ser constante y llevar un registro de ellas [8].

La hembra en celo presenta una temperatura típica comprendida entre 37°C - 39 °C, pero durante el ciclo sexual esta cambia. Antes del estro la temperatura baja, en el momento del estro esta aumenta y durante la ovulación se mantiene [9].

Por otro lado, la temperatura en bovinos (hembra) es considerada un método de detección de estro no visual, ya que no está relacionada con los cambios de conducta, sino con procesos fisiológicos. Durante el estro esta variable aumenta entre 0,3 y 1,1 °C, por lo tanto, se debe tomar constantemente muestras de la misma, que permitan percibir la existencia de estos pequeños cambios mínimos pero significativos que permitan predecir el estado de estro en el bovino [7].

En algunas ocasiones, especialmente en épocas frías, se puede observar como la parte trasera de la vaca emite vapor, esto debido a la elevación de temperatura corporal unido al aumento de la actividad física o los cambios fisiológicos que se presentan a causa del celo [9].

2.3 POSICIÓN GEOGRÁFICA

La posición geográfica se puede determinar a través de GPS donde sus siglas significan Sistema de Posicionamiento Global, este permite determinar la localización de cualquier objeto, y a partir de esta información se pueden obtener otras variables derivadas como son la velocidad, aceleración o la altura [13].

Cuando el bovino se encuentra en estado de celo, uno de los signos característicos es el aumento de la actividad física, es por esto que la posición geográfica es relevante ya que a partir de esta se pueden obtener otras variables derivadas como son la velocidad y aceleración, que pueden ser de útiles para el análisis y estimar el comportamiento del animal.

2.4 FUNCIONAMIENTO DE LOS GPS

La tecnología GPS, fue diseñada por el gobierno de los Estados Unidos alrededor de la década de los años 60 con el fin de reemplazar el sistema Transit, primer sistema de navegación por satélite en funcionar, utilizado en su momento en barcos y submarinos de la marina estadounidense que era poco preciso, arrojando así un error de ubicación con aproximadamente 250 metros, posteriormente, en la década de los años 70 la evolución tecnológica logró obtener un mejor desarrollo y con mayor precisión. Ya en el año de 1983 fue utilizado por el ejército de los estados unidos para incorporarlo en sus distintos equipos y vehículos, siendo liberada esta tecnología en el año 2000 para poder incorporarla en diferentes equipos a nivel mundial [14].

Este sistema funciona mediante una red de alrededor de 24 satélites que se encuentran en la órbita de la tierra, a una distancia aproximada de 20.000 Km de altura, cuando se requiere determinar la posición exacta de un objeto, se necesitan como mínimo 4 de estos satélites, desde donde se envía una señal al receptor, calculando el tiempo de ida y regreso de ella, con el fin de calcular la distancia y la velocidad mediante el método de trilateración inversa. En cuanto al dispositivo como tal, el GPS está compuesto por tres elementos, el usuario, el control y el espacial [13].

Los protocolos más comunes en los dispositivos GPS en la actualidad se listan en la siguiente tabla, ver Tabla 2.

*Tabla 2.
Protocolos GPS*

Protocolos GPS		
Protocolo	Descripción	Características
NMEA	Permite la comunicación entre diferentes dispositivos de navegación, existe actualmente el NMEA 0183 (El más usado) y el NMEA 2000 es la renovación del anterior, con la diferencia que es más rápido en la transmisión, los datos que se transmiten están en formato ASCII y incluye información como velocidad, profundidad, posición [15], [16].	<ul style="list-style-type: none">• Velocidad en baudios: 4800 o 9600• Número de bits de datos: 8 bits• Paridad: Ninguna• Establecimiento de comunicación: Ninguno
SIRF	Permite el cambio de una mayor cantidad de datos, estos datos son mucho más detallados que los que proporciona NMEA, tiene tiempos menores de adquisición de datos, al igual que es más veloz al calcular la posición. La transmisión de datos es en formato binario [17].	<ul style="list-style-type: none">• Velocidad en baudios: 57600• Número de bits de datos: 8 bits• Paridad: Ninguna• Bits de Parada: 1

TSIP	Es un protocolo binario con una velocidad binaria alta, este protocolo permite el intercambio de una gran cantidad de datos y lo hace de manera bidireccional con el receptor [16], [18].	<ul style="list-style-type: none"> ● Velocidad en baudios: 9600 ● Número de bits de datos: 8 bits ● Paridad: Impar ● Bits de parada: 1
------	---	--

2.5 FUNCIONAMIENTO DE LOS PROTOCOLOS DE COMUNICACIÓN GPS

Se trata de un código de señales habilitado para transmitir y recibir datos de diferentes equipos. De esta manera, cuando un dispositivo opera bajo el estándar NMEA-0183, interpreta el formato de los mensajes de este protocolo e identifica el equipo y la información que recibe [19].

La transmisión se realiza en un solo sentido, a partir de un único emisor a uno o más receptores, todos los datos son transmitidos en formato ASCII de 7 bits, donde cada campo se separa con una coma, dichos campos contienen la información de posición, velocidad, profundidad, altura, hora, fecha, entre otros [20].

2.6 DECODIFICACIÓN DE LOS PROTOCOLOS GPS

Todos los datos son transmitidos en sentencias con caracteres ASCII, cada una comienza con el símbolo '\$' y termina con 'CR o LF'. Los siguientes 2 caracteres después de '\$' son los que identifican al equipo, 'GP' (por ejemplo: se utiliza para identificar datos GPS), los tres siguientes caracteres que le siguen son el identificador del tipo de sentencia que se envía. Los tipos de sentencias NMEA que existen corresponden a envió, origen del equipo y consulta [21].

Existen dos tipos de datos que son los más utilizados al momento de recibir los datos GPS, estos son \$GNRMC y \$GNGGA, se configuran por comandos AT [22].

2.6.1 TRAMA \$GNRMC

Esta trama está compuesta por datos relevantes como: Hora, posición, rumbo y velocidad. Además, cuenta con separadores reglamentarios que normalmente son comas. En el siguiente ejemplo se explica cada uno de los elementos que conforman esta trama [22].

Ejemplo:

\$GNRMC,203544.000,A,0438.9198,N,07404.3962,W,0.00,247.85,150517,,,A*62

- **\$GNRMC** : Tipo de dato que se está captando
- **203544.000** : Hora en coordenadas universales, 20 Horas, 35 Minutos, 44.00 Segundos
- **A**: Estado de conexión, Activa (A), sin señal (V).

- **0438.9198, N, 07404.3962, W**: Ubicación en latitud y longitud, Latitud 4 grados y 38.9198 minutos en el hemisferio norte, la longitud 74 grados y 04.3962 minutos en el hemisferio oeste.
- **0.00** : Velocidad en nudos
- **247.85**: Angulo de seguimiento para estimar posiciones siguientes.
- **150517,,,A*62** : Indica la fecha de la forma DD/MM/AA y la variación magnética.

2.6.2 TRAMA \$GNGGA

Esta trama incluye datos como: Hora, posición e información relacionada con el receptor. Además, está compuesta por separadores reglamentarios que usualmente son comas. En el siguiente ejemplo se explica cada uno de los elementos que conforman esta trama [22].

Ejemplo:

\$GNGGA,203544.000,0438.9198,N,07404.3962,W,1,10,0.94,2597.0,M,3.4,M,,*69

- **\$GNGGA** : Tipo de dato que se está captando
- **203544.000,0438.9198,N,07404.3962,W** : Ubicación en latitud y longitud
- **1** : Indicador de calidad, es decir qué tipo de corrección han obtenido los datos GPS.
- **10** : Indica la cantidad de satélites que hay en seguimiento.
- **0.94** : Dilución horizontal de la posición.
- **2597.0,M** : Indica la altitud en metros sobre el nivel del mar
- **3.4,M** : indican una altitud relativa de la ubicación.

2.7 TECNOLOGÍA DE COMUNICACIÓN IOT

El sector de IoT ha ido evolucionando satisfactoriamente en los últimos años, generando que actualmente existan multitud de redes de comunicación asociadas a esta tecnología. Entre las más utilizadas se encuentra: Sigfox, LoRaWAN, ZigBee, Wifi, entre otras. Dependiendo de la aplicación y las características requeridas (alcance, consumo, frecuencia y demás), se podrá elegir la mejor opción.

A continuación, se hace una reseña de algunas de las tecnologías de comunicación IoT más utilizadas actualmente:

2.7.1 SIGFOX

Es una red de comunicación fiable, de bajo consumo, bidireccional, económica, y de una gran cobertura a nivel global que conecta sensores y dispositivos con mensajes ultraligeros de 12 bytes. Esta tecnología funciona de la siguiente manera [23]:

1. El dispositivo se despierta y emite un mensaje con su antena de radio.
2. Múltiples estaciones base Sigfox en el área reciben el mensaje.

3. Las estaciones base envían el mensaje a la nube de Sigfox.
4. La nube de Sigfox envía el mensaje a la plataforma de backend del consumidor.

A continuación, se muestra la estructura de la red Sigfox, ver Ilustración 2.



Ilustración 2. Estructura red Sigfox. Fuente [23].

2.7.2 LORAWAN

Es una tecnología para comunicar pequeños dispositivos electrónicos empleados en la denominada Internet de las Cosas (IoT), Esta tecnología permite el envío y recepción de información punto-a-punto. Lo que caracteriza a un dispositivo LoRa es su largo alcance [24].

Uno de los inconvenientes que presenta esta red es su cobertura, ya que depende de las empresas que deseen implementarla. LoRaWAN a diferencia de Sigfox no cuenta con un operador global que esté a cargo del despliegue, por lo que son los propios usuarios quienes deben instalar Gateway para ampliar la cobertura [25].

En la Ilustración 3 se representa la estructura de la red LoRaWAN.

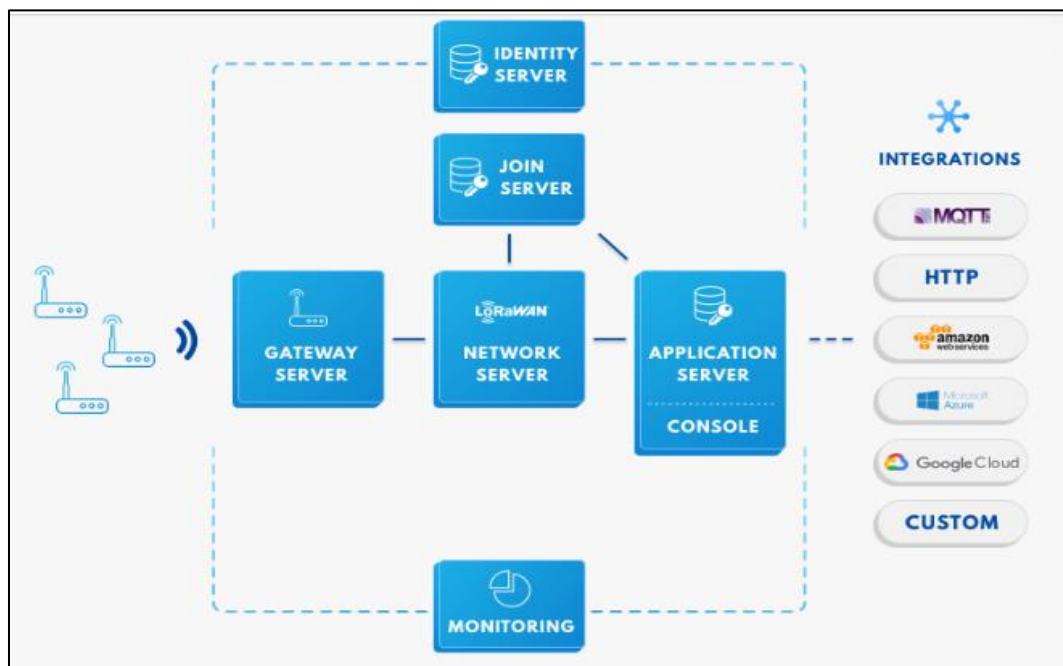


Ilustración 3. Estructura red LoRaWAN. Fuente [26].

2.7.3 ZIGBEE

Es una tecnología inalámbrica que tiene aplicaciones domóticas e industriales, para su implementación se deben instalar y configurar los nodos que se requieran, por lo tanto el alcance depende de cada usuario, pues no existe un operador que se encargue del despliegue, esta red opera con el estándar 802.15.4 de IEEE, donde utiliza la banda de frecuencia de 2.4 GHz [27].

Para entender mejor el funcionamiento de Zigbee se presenta la estructura en la Ilustración 4.

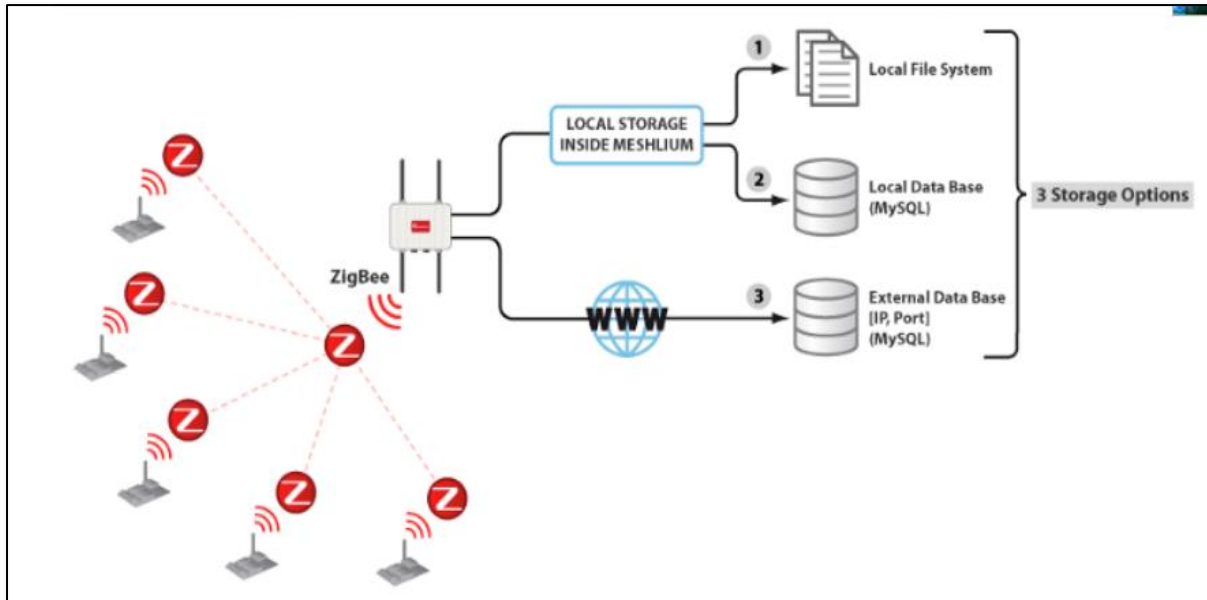


Ilustración 4. Esquema red Zigbee. Fuente [28].

2.7.4 WIFI

Es la tecnología inalámbrica más común en el mundo ya que por ella se transmite gran cantidad del tráfico global de internet, se basa en el estándar de comunicación inalámbrica 802.11, posee la capacidad de transmitir datos a velocidades altas y maneja grandes cantidades de información, actualmente posee alrededor de 13 mil millones de dispositivos en uso lo que la hace la tecnología más utilizada, una desventaja evidente es el alcance, ya que es muy baja en comparación con otras [29].

A continuación, se muestra la estructura de la red wifi, ver Ilustración 5.

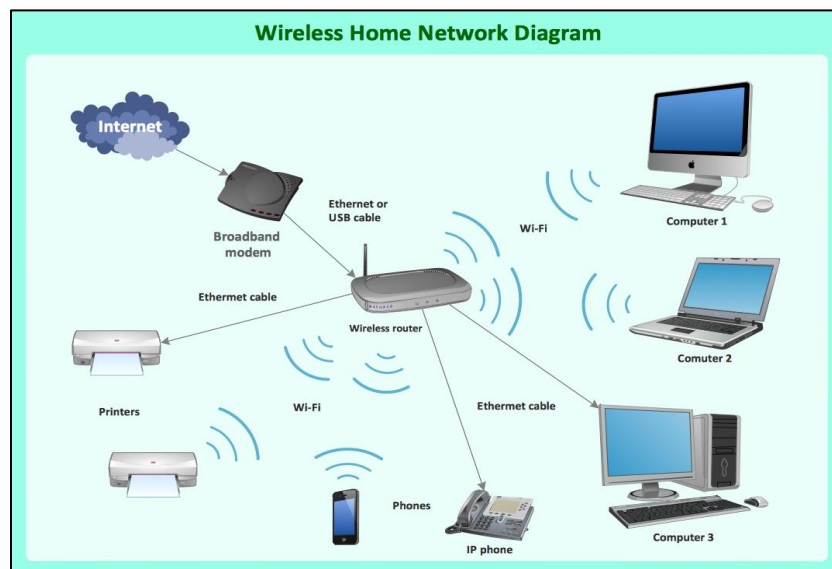


Ilustración 5. Estructura red Wifi. Fuente [30].

A continuación se enlistan algunas características típicas que permiten elegir adecuadamente una red de comunicación IoT [31].

- Bajas velocidades de datos
- Baja frecuencia de transmisión
- Movilidad y servicio de localización
- Conexiones bidireccionales seguras
- Bajo consumo de energía
- Largo alcance de comunicación

3. DISEÑO DEL SISTEMA ELECTRÓNICO

En este apartado se llevará a cabo la planeación del presente proyecto, teniendo en cuenta que para realizar la estimación de estro en un bovino (hembra), es necesario monitorear la variable fisiológica temperatura, por lo que se contempla el diseño de un prototipo electrónico.

Esta sección estará dividida en tres etapas, la primera de ellas es el diseño hardware donde se describen los requerimientos electrónicos, la arquitectura del sistema propuesto, y la elección de los diferentes componentes electrónicos utilizados en el prototipo. La segunda etapa es el diseño software, donde se definen los requisitos iniciales, el firmware, y las herramientas a utilizar como es el caso de las plataformas IoT. Por último, se encuentra la etapa de implementación del sistema electrónico, en este punto se realizarán las diferentes pruebas, mediciones y análisis que necesite el prototipo para cumplir las condiciones iniciales.

3.1 DISEÑO HARDWARE DEL SISTEMA

Para la implementación del prototipo es necesario tener en cuenta algunos aspectos relacionados con diseño y construcción de hardware, por lo tanto, es importante comenzar con la definición de unos requerimientos generales, luego de esto se deben seleccionar los componentes adecuados y finalmente realizar el diseño electrónico de todo el sistema.

3.1.1 DEFINICIÓN DE REQUERIMIENTOS

En este apartado se describen los requerimientos electrónicos iniciales y se justifica cada uno de ellos con el fin de llevar a cabo la realización de todo el sistema electrónico, teniendo así una base referencial para ello, ver Tabla 3.

*Tabla 3.
Requerimientos electrónicos del sistema. Fuente Autor*

REQUERIMIENTOS ELECTRÓNICOS DEL SISTEMA	JUSTIFICACIÓN
1. Que tenga una medida máxima de 83x58x35mm.	Se toma como base de referencia, dispositivos electrónicos comerciales similares, tal como DigitAnimal que tiene un tamaño de 159mm x 115.5mm x 7.5mm [32].
2. Que cuente con un sensor de temperatura con un rango de medida al menos de 30°C - 50°C.	Según la literatura consultada, los valores típicos de temperatura corporal de los animales bovinos varían entre 37.7°C y 39°C [12].

3. Que cuente con un sensor de temperatura con una precisión de 0,5°C y una resolución de 0,02°C aproximada.	De acuerdo a las variaciones de la temperatura cuando el bovino (hembra) se encuentra en estado de estro, estas son del orden de las décimas de grado.
4. Que cuente con una gestión de operación eléctrica de bajo consumo.	Debido a que será un dispositivo utilizado en el campo y portable se requiere baja intervención humana para su funcionamiento.
5. Que cuente con un sistema de comunicación y una base de datos para almacenar la información.	Con el fin de llevar un histórico para poder caracterizar la variable de temperatura del animal.
6. Se debe tener una interfaz software de usuario que permite mostrar información sobre, temperatura, ubicación y sus respectivas gráficas.	Con el fin de visualizar y analizar la información almacenada en la base de datos.
7. Que permita la generación de alertas sobre el sistema.	Para que el usuario esté informado sobre el estado de los animales (bovinos).

3.1.2 DISEÑO ELECTRÓNICO

En este capítulo se presenta de manera general el diseño electrónico de todo el sistema, y se describe cada uno de los bloques que lo conforman, ver Ilustración 6.

El sistema se divide en dos secciones (hardware, software), en el caso de la parte hardware se encuentran los bloques de sensado, control, alimentación y comunicación. ya en la sección de software están los bloques de infraestructura IoT, cloud y cloud computing. A continuación, se describirán más en detalle cada uno de estos bloques.

- A. En el primer bloque del sistema electrónico se presenta la etapa de sensor, donde se realiza la adquisición de los datos de temperatura y ubicación, con esta información se puede conocer el estado del animal y en qué lugar del terreno se encuentra.
- B. En el bloque de control se procesan las señales obtenidas en la etapa de sensado, y se adquieren medidas en grados centígrados para el caso del sensor de temperatura y datos de latitud, longitud para el GPS.
- C. La fuente de energía es un factor fundamental en el sistema, es por esto que se busca una estrategia para el ahorro de energía y así tener un mejor aprovechamiento de los recursos.

- D. El bloque de comunicación inalámbrica IoT recibe los datos adquiridos por la etapa de control y posteriormente los envía a la nube por medio de de módulos de comunicación IoT.
- E. La infraestructura IoT se encarga de codificar y decodificar la información enviada por medio de antenas y servidores para así luego ser mostrada en internet.
- F. La etapa de Cloud es aquella que obtiene la información para así almacenarla y posteriormente enviarla a una plataforma de visualización en la nube.
- G. Finalmente, en Cloud Computing se podrán analizar los datos según los requerimientos iniciales del proyecto y así permitirle al usuario visualizar toda la información relevante.

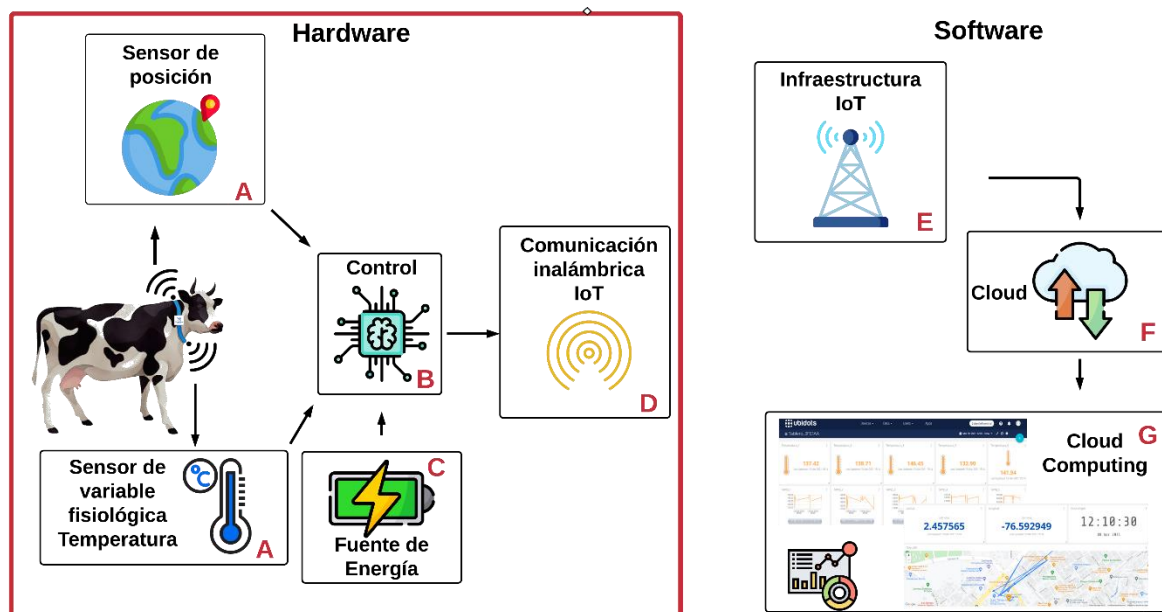


Ilustración 6. Plataforma de monitoreo IoT. Fuente Autor.

3.1.3 SELECCIÓN DE COMPONENTES

Para que el sistema opere de acuerdo a las especificaciones propuestas es importante realizar una selección de los componentes que se van a utilizar, se debe tener en cuenta diferentes factores eléctricos tales como: consumo, tamaño, rangos de medición, señal de salida y demás.

3.1.3.1 SENSORES DE TEMPERATURA





Estos dispositivos de medición son utilizados en diferentes procesos, pues con ellos se determina la temperatura del ambiente o la de un objeto, y a partir de esta lectura

se realizan distintos análisis. Cabe mencionar que existen variedad de elementos para medir la temperatura y así mismo diferentes parámetros al momento de elegir un sensor de este tipo.

Para la elección del sensor de temperatura se tendrá en cuenta características tales como: precisión, resolución, calibración, rango de medida, voltaje de alimentación, entre otros, con el fin de elegir el sensor que más se acomode a nuestras necesidades [33].

En la Tabla 4 se muestra la comparación entre distintos sensores comerciales de temperatura, que cumplen con los requisitos planteados para el presente proyecto.

Tabla 4.
Sensores de temperatura. Fuente [34], [35], [36], [37].

SENSORES DE TEMPERATURA				
SENSOR	SI7051-A20- IMR 	TSYS01 	MLX90614 	DS18B20+ 
VOLTAJE DE ALIMENTACION	1.9 - 3.6V	2.2 – 3.6V	3 a 3.6V	3 a 5.5V
CONSUMO	195nA	12,5uA	1.3mA	1mA
SEÑAL DE SALIDA	I ² C	I ² C - SPI	I ² C	1-Wire
RANGO DE MEDIDA	-40 – 125 °C	-40 – 125 °C	Amb:-40 a 125 °C Obj: -70 a 380 °C	-55 a 125 °C
PRECISIÓN	±0.1°C (±0.25°C)	±0.1°C (± 0.5)	±0.5 °C	±0.5°C
RESOLUCIÓN	0.08 - 0.01 °C	0.01 °C	0.02 °C	0.5 - 0.0625 °C
CALIBRACION DE FABRICA	Si	Si	Si	No
TAMAÑO	3mm x 3mm (6-WDFN)	4mm x 4mm (16- VQFN)	9.12mmX17.10mm (TO-39)	5.2mmX4.2mmX19.4mm (TO-92)

COSTO	\$7.300	\$17.100	\$30.000	\$8.330
--------------	---------	----------	----------	---------

De los anteriores dispositivos se opta por utilizar el sensor MLX90614 debido a las siguientes razones:

- Gracias a su tamaño permite facilidad en su montaje pcb.
- Al ser un sensor infrarrojo cuenta con una alta precisión.
- Permite la medición de dos temperaturas (ambiente, objeto), diferenciándolas entre sí.
- Su señal infrarroja permite medir la temperatura sin necesidad de contacto.





3.1.3.2 SENSORES DE GPS

Estos componentes proporcionan la ubicación geográfica lo más aproximada posible de forma latitudinal y longitudinal, comúnmente son utilizados en celulares, transporte (Aéreo, terrestre, fluvial), animales, personas y demás. Existen variedad de dispositivos GPS que se usan según su fin: seguridad, rastreo, salud, ingeniería, entre otros [22].

Para la elección del GPS se tendrá en cuenta los siguientes parámetros: rango de medida, precio, voltaje de alimentación, entre otros. Para así determinar el sensor GPS más adecuado.

En la Tabla 5 se presenta la comparación entre distintos GPS comerciales, que cumplen con los requisitos planteados para el proyecto.

*Tabla 5.
Sensores de GPS. Fuente [38], [39], [40], [41].*

Sensores GPS					
Sensor					
Voltaje de Alimentación	3 – 4.3 V	3 – 4.3 V	1.6 – 3.6V	2.8 – 4.3 V	
Canales	Rastreo	22	33	72	
	Adquisición	66	99		
Tamaño	15 mm x 15 mm x 4mm	18.4 mm x 18.4 mm x 6.45 mm	9.6 mm x 14 mm x 2 mm	14 mm x 9.6 mm x 2.5 mm	
Costo	\$25.000	\$34.510	\$93.250	\$33.320	
Consumo	Rastreo	20 mA	22mA	23mA	20 mA
	Adquisición	25 mA	25 mA	26mA	22 mA

Precisión de Posición	< 2.5 m	< 2.5 m	<2.5 m	<2.5 m
Protocolo	NMEA 0183 / PMTK	NMEA 0183 / PMTK	NMEA 0183 / UBX / RTCM	NMEA 0183 / PMTK

De los anteriores dispositivos se escogió el sensor GPS L80-R de Quectel por las siguientes razones:

- Por su bajo consumo de corriente.
- Variedad de información sobre el dispositivo.
- Tiene antena embebida.
- Cuenta con alta precisión y sensibilidad.
- Tamaño compacto.

3.1.3.3 MICROCONTROLADOR

Este dispositivo al ser programado cumple la función de controlar tareas determinadas dentro de un sistema y está compuesto por las siguientes partes: una unidad de procesamiento central (CPU), una memoria para almacenar programas, una memoria para almacenar datos y puertos (entrada, salida). Los microcontroladores son ampliamente utilizados en diversos campos (comunicaciones, aparatos electrónicos, robótica y demás) [33].

Las principales características a tener en cuenta a la hora de elegir el microcontrolador son: bajo consumo, que cuente con numerosos pines I/O, que tenga RTC (Reloj en tiempo real) integrado el cual ayuda a mantener la hora actual y a su vez permite tener un bajo consumo de energía.

En la Tabla 6 se muestra la comparación de diferentes tipos de microcontroladores que cumplen con los requerimientos del proyecto.

*Tabla 6.
Microcontroladores. Fuente [42], [43], [44], [45].*

Microcontroladores				
MCU	ATSAMD21G18	ATmega328P	PIC24F08KA101	PIC24f08KM204
Voltaje de Operación	1.62 V – 3.63 V	2.7 V – 5.5 V	1.8V - 3.6 V	1.8 V – 3.6 V
Consumo de corriente	3.9mA	2.4mA	8 uA	13mA
Consumo de corriente en modo sueño	15 uA	44 uA	25nA	25nA
SPI	6	2	1	2
I2C	6	1	1	2
Pines I/O	48	32	20	44
SRAM	32	2	1.5	2

[kB]				
Flash [kB]	256	32	8	8
Hardware RTCC / RTC	Si	No	Si	Si

El microcontrolador elegido es el ATSAM21G18 de Atmel, ver Ilustración 7, ya que este dispositivo cuenta con un gran potencial y se puede utilizar con la interfaz de programación arduino, lo que hace más rápida la curva de desarrollo, por otro lado cuenta con seis módulos de comunicación en serie (SERCOM) que pueden ser configurados como UART/USART, SPI, I2C según sea la necesidad [42].

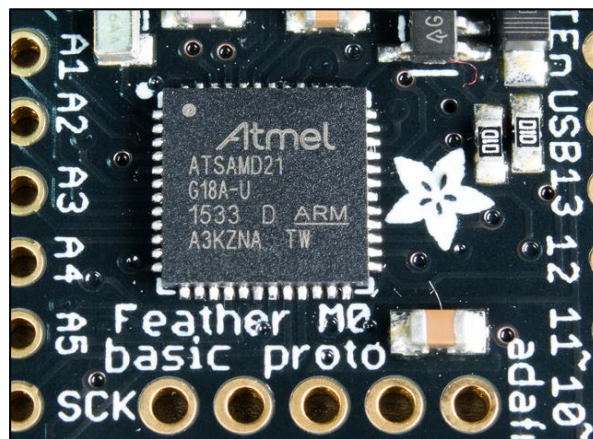


Ilustración 7. Microcontrolador ATSAM21G18A. Fuente [46].

3.1.3.4 SELECCIÓN DE SD

El prototipo requiere de un sistema de respaldo en caso de que la comunicación con Sigfox falle y por ende haya pérdida de datos. Por esta razón se opta por usar una tarjeta SD que permite guardar la información en un archivo “.CSV” donde los datos estarán seguros para su posterior análisis [47].

Para su funcionamiento es necesario un socket o ranura el cual permite la conexión entre el microcontrolador y la tarjeta SD. El socket utiliza el protocolo SPI (Interfaz periférica serial), que cuenta con un único maestro y los demás dispositivos se comportan como esclavos, por otro lado posee 4 señales (MOSI, MISO, SCK y CS) para comunicarse con él microcontrolador, ver Ilustración 8.

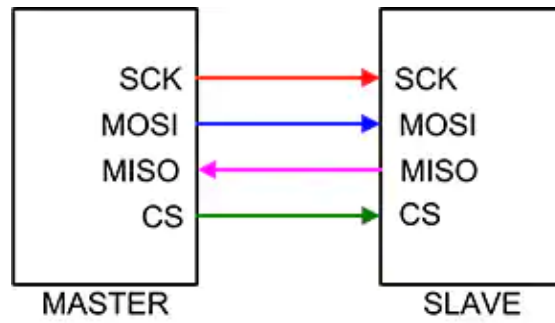


Ilustración 8. Comunicación SPI. Fuente [48].

En la Tabla 7, se muestra la clasificación de las tarjetas SD según su tamaño, almacenamiento y velocidad. En el mercado hay diferentes tipos, por lo que se tendrá en cuenta las más comunes [49].

Tabla 7.
Clasificación tarjeta SD. Fuente [49].

Clasificación Tarjetas SD		
Tamaño	Nombre	Dimensiones en mm (alto x ancho)
	Tarjeta SD	32 x 24
	Tarjeta Mini SD	21.5 x 20
	Tarjeta Micro SD	15 x 11
Almacenamiento	Tipo de tarjeta	Capacidad
	SD (Capacidad Estándar)	1 MB - 2 GB
	SD HC (Capacidad Alta)	2 GB – 32 GB
	SD XC (Capacidad Extendida)	32 GB - 2TB
Velocidad	Tipo	Velocidad
	Clase 2	2 MB/s
	Clase 4	4 MB/s
	Clase 6	6 MB/s
	Clase 10	10 MB/s
	UHS Clase 1 (U1)	10 MB/s
	UHS Clase 2 (U3)	30 MB/s

La tarjeta seleccionada es la micro SD tipo HC, por su tamaño reducido (15 x 11 mm), capacidad de almacenamiento (4GB) y velocidad (clase 4).

3.1.3.5 SELECCIÓN FUENTE DE ALIMENTACION

Todo dispositivo electrónico necesita una fuente de alimentación para su funcionamiento, por lo tanto, es necesario analizar qué tipo de fuente se acomoda más a los requerimientos electrónicos planteados en este proyecto. Para la elección se tendrán en cuenta factores como: tamaño, capacidad, peso, tensión, tiempo de

carga y ciclo de carga. En la Tabla 8 se muestra la comparación de algunos tipos de fuentes de alimentación.

*Tabla 8.
Características de la batería. Fuente [50].*

Características de las baterías					
Tipo	Voltaje	Costo	Mantenimiento	Ciclo de carga	Tiempo de Carga
Plomo (Pb)	2 V	Bajo	Muy alto	500 - 2000	8-16 Horas
Níquel-Cadmio (Ni-Cd)	1.2 V	Moderado	Elevado	500 - 1000	10-14 Horas
Níquel-Hidruro (Ni-MH)	1.2 V	Muy Alto	Bajo	500- 800	2-4 Horas
Alcalino	1.5 V	Muy Bajo	Bajo	Bajo	NA
Iones de Litio (Li-ion)	3.7 V	Muy Alto	Bajo	1000 - 1200	2-4 Horas
Polímero de Litio (Li-Po)	3.7 V	Muy Alto	Bajo	>1000	1-1.5 Horas

La fuente de alimentación elegida es la batería lipo por las siguientes razones:

- Tamaño reducido que se adapta a las medidas que tiene la caja del prototipo.
- Menor tiempo de carga
- Voltaje de celda necesario para suministrarle a todo el circuito
- Larga vida útil gracias a que posee más ciclos de carga
- No requiere mantenimiento.

Teniendo el tipo de batería seleccionado, se procede a calcular el consumo de todo el circuito, para así determinar la capacidad que debe tener la batería lipo elegida. La Tabla 9 , contiene el consumo en modo normal y modo sueño de cada uno de los componentes del sistema.

*Tabla 9.
Consumo de corriente del sistema. Fuente[42], [51], [36], [38].*

Dispositivo	Consumo de corriente - modo normal	Consumo de corriente – modo transmisión	Consumo de corriente - modo dormido
Microcontrolador (ATSAMD21G18)	3.9 mA	3.9 mA	0.015mA
Módulo Wisol	32mA	170mA	0.0024mA
Modulo GPS (L80-R)	20mA	25mA	1mA
Sensor de Temperatura MLX90614	1.3mA	1.3mA	0.006mA
Modulo SD	200mA	-	-
Total	257.2mA	200.2mA	1.0234mA

Una vez presentados los datos de consumo, se definen los valores totales de corriente y tiempo, ver Tabla 10.

Tabla 10.
Variables para calcular el consumo energético. Fuente Autor.

Tiempos		
Nombre	Intervalos (Minutos y segundos)	Intervalos (Horas)
Tiempo de consumo modo normal (Tcn)	30s	0.00833333 h
Tiempo de consumo modo transmisión (Tct)	2s	0.000555556 h
Tiempo de consumo modo sleep (Tcs)	10 Min	0.166667 h
Corrientes		
Nombre	Consumo	
Consumo de corriente - modo normal (Ccn)	257.2 mA	
Consumo de corriente - modo transmisión (Cct)	200.2 mA	
Consumo de corriente - modo sleep (Ccs)	1.0234 mA	

A continuación se calcula el consumo total de todo el circuito [52].

$$consumo = \frac{Tcn * Ccn + Tct * Cct + Tcs * Ccs}{Tcn + Tct + Tcs} \quad (\text{Ecuacion 1})$$

$$consumo = \frac{0.00833333 h * 257.2 mA + 0.000555556 h * 200.2 mA + 0.166667 h * 1.0234 mA}{0.00833333 h + 0.000555556 h + 0.166667 h} \quad (\text{Ecuacion 2})$$

$$consumo = 13.81mA \quad (\text{Ecuacion 3})$$

Teniendo el consumo eléctrico del dispositivo se opta por utilizar una batería lipo de 850mAh ya que su capacidad y tamaño se acomodan a los requerimientos de diseño del prototipo. Para calcular el tiempo de duración de la batería se usa la siguiente fórmula.

$$\text{Tiempo de duracion de la bateria} = \frac{\text{Carga electrica}}{\text{Consumo electrico del dispositivo}} \quad (\text{Ecuacion 4})$$

$$\text{Tiempo de duracion de la bateria} = \frac{850mAh}{13.81395891mA} \quad (\text{Ecuacion 5})$$

$$\text{Tiempo de duracion de la bateria} = 61.5319624 \text{ horas} \quad (\text{Ecuacion 6})$$

El tiempo de duración estimado para la batería de 850mA, es de 2 días 13 horas 31 minutos y 55 segundos, ver Ilustración 9.



Ilustración 9. Batería LiPo 850 mAh. Fuente[53].

3.1.3.6 SELECCIÓN TECNOLOGÍA DE COMUNICACIÓN PARA IOT

Para la selección de la tecnología IoT se tienen en cuenta factores como: el costo que pueda representar el dispositivo, la facilidad de transmisión de información a la nube, el consumo de corriente usado en los transceptores, entre los más destacados, adicional a esto debe cumplir con los requerimientos electrónicos planteados en el presente proyecto.

A continuación, se muestra una tabla comparativa con algunas de las tecnologías asociadas a IOT, Ver Tabla 11.

Tabla 11.
Tecnologías asociadas a IoT. Fuente [31], [52], [54], [55], [56], [57].

TECNOLOGÍA	Sigfox	LoRaWAN	ZigBee	Wifi
CONSUMO	Bajo Tx: <50mA Rx: 10-40mA Sleep: <0.01 mA	Bajo Tx: <50mA Rx: 10-40mA Sleep: <0.01 mA	Medio Tx: 52mA Rx: 54mA Sleep: 12 uA	Alto Tx: 251mA Rx: 248mA Sleep: 30uA
ALCANCE	30 – 50Km (Ambientes rurales)	15 Km (entorno rural)	10 – 100 m	Aproximadamente 50 m

FRECUENCIA	920 MHz	433 - 434.79 MHz 915 - 928 MHz	2.4GHz	2.4GHz y 5GHZ
VELOCIDAD DE TRANSFERENCIA	10 – 1000 bps	0,3 – 50 Kbps	250 Kbps	600 Mbps
SEGURIDAD	Media	N A	Baja	Alta
USO DE GATEWAYS	No	Si	Si	No

Sigfox se escogió como la tecnología de comunicación por las siguientes razones: [31].

- Bajo consumo de energía, lo que genera un largo tiempo de vida a la batería.
- Largo alcance de comunicación.
- Amplia cobertura en el departamento del Cauca, lugar donde se realiza la investigación [58], ver Ilustración 10.
- Bajo costo en cuanto a la adquisición de dispositivos y suscripción de sus servicios.
- Comunicación de datos bidireccional.
- Mensajes cortos 12 Bytes Uplink y 4 Bytes Downlink.
- La curva de aprendizaje y manejo es más corta debido a que es una tecnología P2P, por lo que no necesita de montaje, ni configuración de infraestructura como torres o Gateway para tener los datos en la nube.

Sigfox service maps

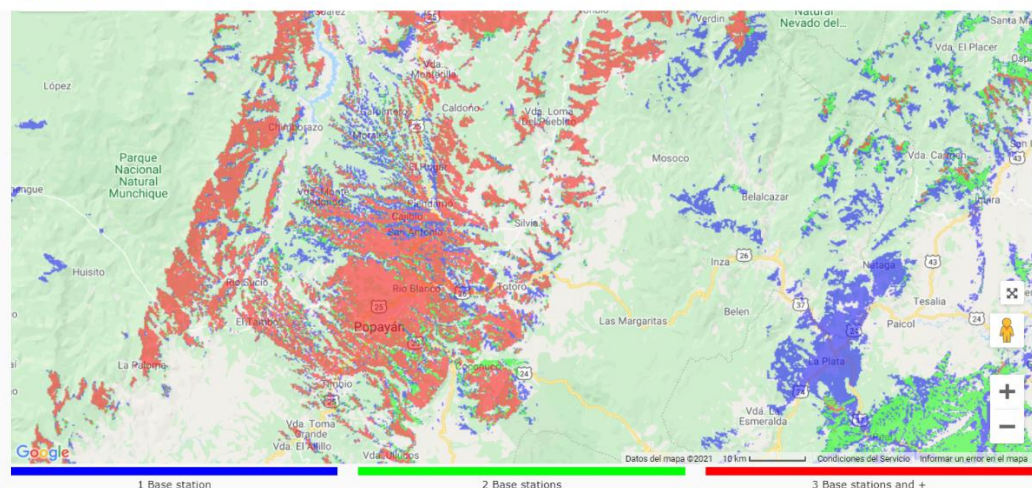


Ilustración 10. Mapa de cobertura Sigfox. Fuente[58].

3.1.3.6.1 SELECCIÓN DEL TRANSECTOR DE COMUNICACIÓN SIGFOX

Una vez seleccionada la tecnología de comunicación IoT se requiere de un transceptor para ser integrado en el sistema hardware. Estos componentes electrónicos permiten la comunicación con la nube de Sigfox, es decir, los transceptores se encargan del envío y recepción de datos, entre sus características más destacadas se encuentra su bajo consumo energético, buen rendimiento y gran alcance. Cabe resaltar que estos dispositivos son utilizados en diferentes campos como la industria, agricultura, salud, seguridad, hogar, entre otros [59].

Para la elección de este dispositivo es necesario que opere en la zona cuatro (RC4), ya que puede ser usado en la red de Sigfox Colombia. De igual forma es importante que el transceptor maneje el protocolo de comunicación serial, pues este permite controlarlo a través de comandos AT [60].

A continuación, en la Tabla 12 se listan algunos transceptores compatibles con Sigfox que se encuentran en el mercado.

Tabla 12.
Transceptores. Fuente [51], [61], [62], [63].

TRANSECTOR	SFM11R2D	IMCP HT32SX	SN10-11	SN10-12
VOLTAJE DE ALIMENTACION	2.7 V - 3.6 V	2.7 V - 3.6V	1.9V – 3.6V	3.3V – 3.6 V
PROTOCOLO DE COMUNICACION	UART	USART, LPUART,USB2.0,I2C	SPI	SPI
CONSUMO DE CORRIENTE	TX – 170mA	TX – 166.5mA	TX – 33mA	TX – 260mA
	RX – 32mA	RX – 17.7mA	RX – 14mA	RX – 17.5mA
	IDLE - 2.5uA	IDLE - 8uA	IDLE – 2.5uA	IDLE – 4.4uA
POTENCIA DE TRANSMISION	+24dBm	+22dBm	14dBm	+23dBm
ZONAS DE COBERTURA	RC2, RC4	RC1, RC2, RC3, RC4, RC5, RC6, RC7	RC1	RC2, RC4
COSTO	\$ 24.752	\$ 168.931	\$ 14.630	\$ 17.027

El transceptor elegido es el SFM11R2D por los motivos que serán nombrados a continuación:

- Maneja protocolo de comunicación serial, puede ser controlado a través de comandos AT.
- Está habilitado para trabajar en zona cuatro (RC4), lo que permite ser usado en la red de Sigfox Colombia.
- El voltaje de alimentación se acomoda a los requerimientos propuestos en el presente proyecto.
- Bajo consumo de energía con relación a los dispositivos comparados en la anterior tabla.

3.1.4 DISEÑO DEL PLANO ESQUEMÁTICO

Teniendo definido el diagrama de la plataforma de monitoreo IoT (ver Ilustración 6), y los componentes seleccionados, se presenta el diagrama en bloques del circuito electrónico de todo el sistema (ver **¡Error! No se encuentra el origen de la referencia.**). A continuación, se describe cada bloque.

1. El bloque de sensores se encarga de la medición de parámetros corporales y geográficos, para este caso como se mencionó en anteriores capítulos los sensores seleccionados son, el sensor infrarrojo de temperatura MLX90614 y el módulo GPS L80-R.
2. Se recolectan y procesan las señales medidas por los sensores, por medio del microcontrolador Samd21. La información obtenida es enviada por comunicación serial al módulo de Sigfox
3. En la etapa de comunicación el módulo de Sigfox, cumple la función de recibir los datos enviados por el microcontrolador y posteriormente transmitirlos al backend de Sigfox.
4. Este bloque permite el ahorro de consumo, ya que el switch cumple la tarea de apagar o encender el módulo Sigfox y GPS en el momento que se requiera.
5. Permite la alimentación y autonomía de todo el sistema electrónico, para suministrarle 3.3 V a todo el circuito.

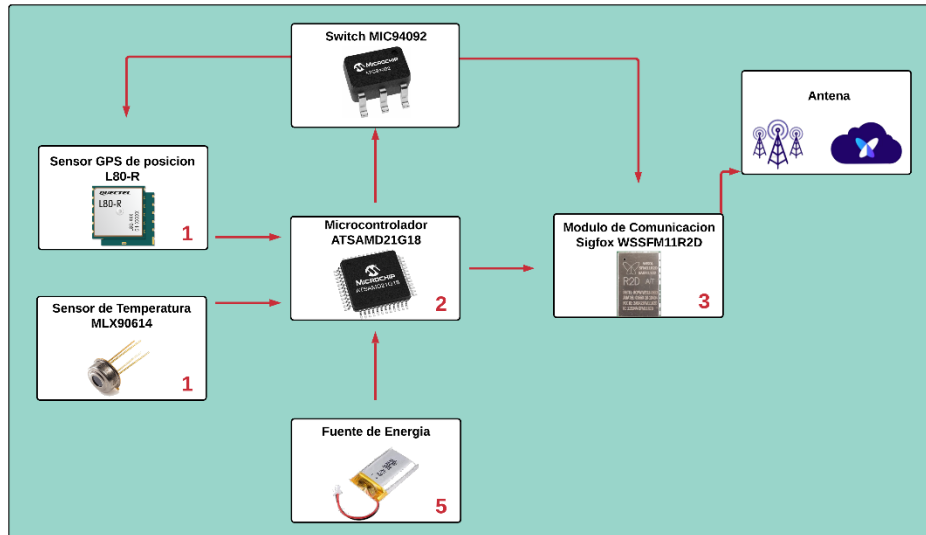


Ilustración 11. Diagrama en bloques del circuito electrónico. Fuente Autor.

Una vez elaborado el diagrama en bloques del circuito electrónico (ver **¡Error! No se encuentra el origen de la referencia.**), se procede a realizar el plano esquemático del prototipo, para esto se hace uso del software EAGLE [64], que permite diseñar placas de circuito impreso (PCB), ubicar componentes y enrutar.

En el diagrama esquemático de la Ilustración 12, se encuentran marcados en bloques los diferentes componentes principales que conforman el prototipo, en el primer bloque están los sensores (temperatura, GPS, acelerómetro) con sus respectivos elementos de acondicionamiento, seguidamente de este recuadro se encuentra el microcontrolador, los módulos (Sigfox, Sd) y por último la etapa de alimentación de todo el sistema.

Para él diseño se contempla la integración de un sensor inercial que será dejado a disposición para trabajos futuros ya que permite adquirir variables importantes como posición, velocidad, aceleración, velocidad angular, inclinación, sin embargo, para él presente trabajo de grado no se tomará en cuenta debido a la complejidad en él desarrollo de los algoritmos asociados a unidades inerciales.

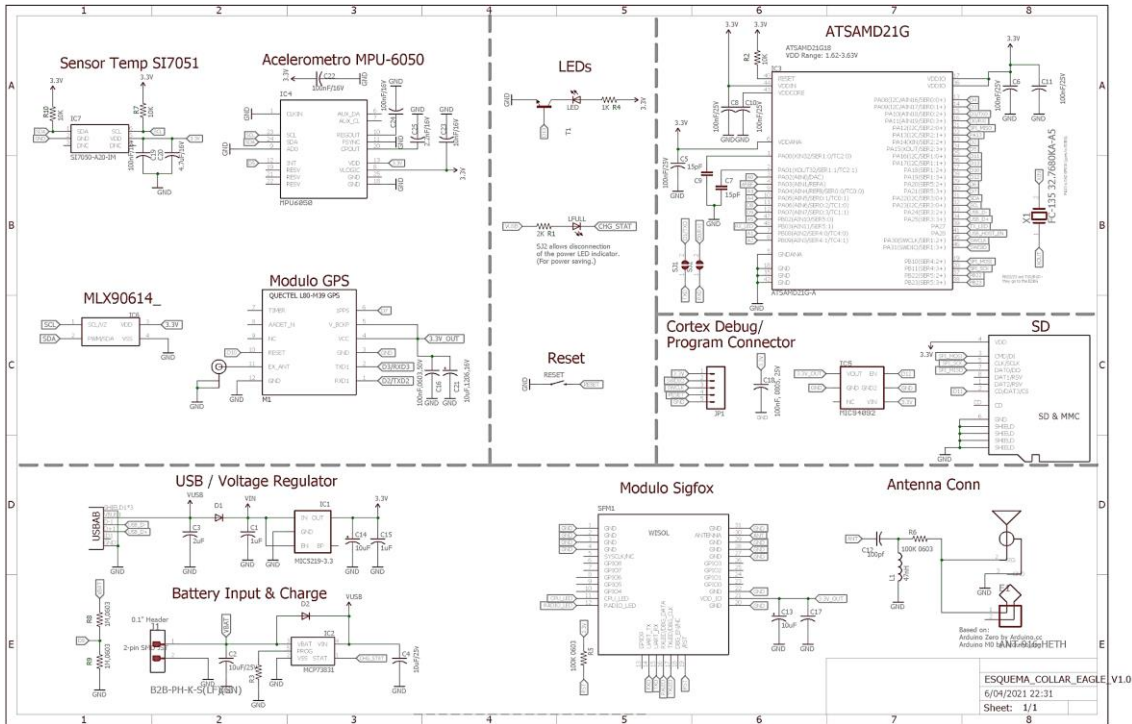


Ilustración 12. Diseño Electrónico en Eagle (SCH). Fuente Autor.

A continuación se muestra el ruteo de todo el diseño electrónico del sistema, en formato .BRD. Ver Ilustración 13.

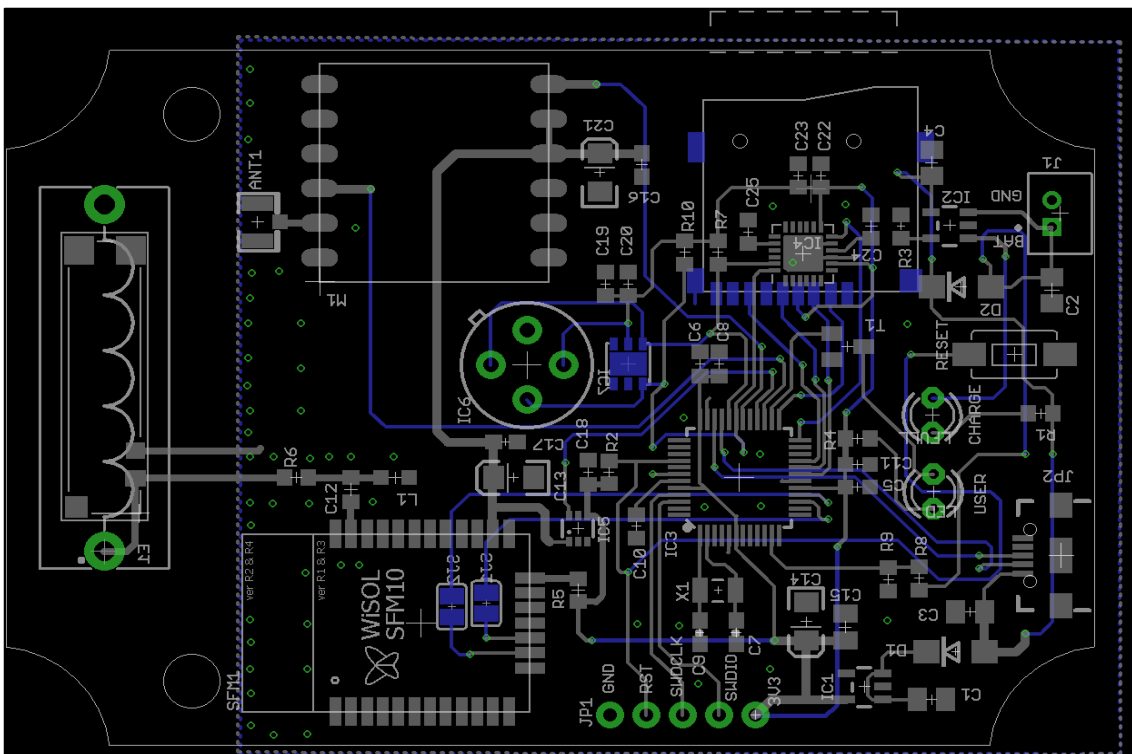


Ilustración 13. Diseño electrónico en Eagle (BRD). Fuente Autor.

Finalmente, en la Ilustración 14 se visualiza la tarjeta el pcb ensamblada, donde se señala cada una de las partes que la conforman.

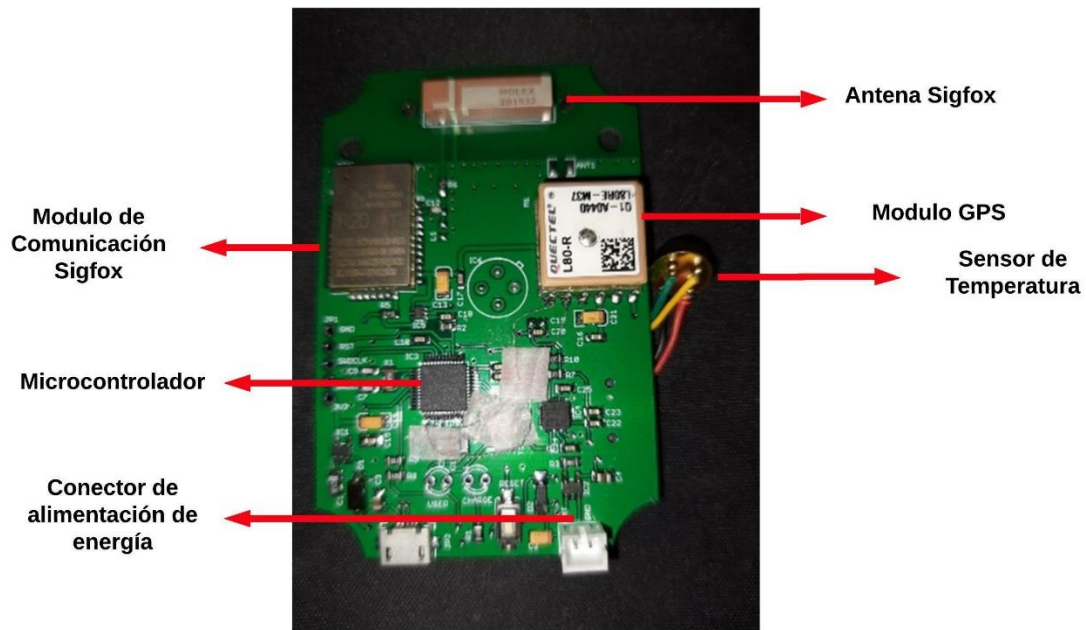


Ilustración 14. PCB Ensamblado. Fuente Autor.

3.1.5 ESTRATEGIA ELECTRONICA DE CONTROL PARA BAJO CONSUMO

Como se observó en anteriores capítulos, el tiempo que dura el sistema funcionando es de 2 días aproximadamente, por lo que es necesario desarrollar una estrategia electrónica de bajo consumo, para que la batería tenga un tiempo de descarga mayor y con esto el prototipo funcione por más tiempo.

Para obtener un menor consumo energético, se utilizan conjuntamente dos herramientas, la primera de ellas consiste en la librería RTCZero capaz de poner en modo suspensión el microcontrolador a través de sentencias y alarmas previamente programadas [65]. El modo suspensión logra reducir el consumo ya que apaga algunos periféricos del microcontrolador que no son necesarios en el momento.

La segunda técnica es la utilización de un switch que funciona como un controlador para apagar o encender los módulos GPS Y WISOL en el momento que se requiera.

Para llevar a cabo la estrategia de control se hace lo siguiente:

- Poner en modo suspensión (Standby) al microcontrolador con esto se garantiza que el consumo baje considerablemente.
- Controlar a través de un switch los módulos Wisol y GPS, logrando así que solo consuman corriente en el momento que se requiera.

A continuación, se presenta la Tabla 13 con el consumo de cada uno de los componentes, de igual forma se presentan los distintos modos de operación.

*Tabla 13.
Consumo eléctrico del sistema - Estrategia bajo consumo. Fuente[36], [38], [42], [51].*

Dispositivo	Consumo de corriente - modo dormido (i1)	Consumo de corriente - modo transmisión temperatura (i2)	Consumo de corriente - modo normal-adquisición GPS (i3)	Consumo de corriente - modo transmisión GPS (i4)
Microcontrolador(ATSAMD21G18)	0.015mA	3.9 mA	3.9 mA	3.9 mA
Módulo Wisol	0	170mA	32mA	170mA
Modulo GPS (L80-R)	0	20mA	20mA	25mA
Sensor de Temperatura MLX90614	0.006mA	1.3mA	1.3mA	1.3mA
Modulo SD	-	-	-	-
Total	0.021mA	195.2mA	57.2mA	200.2mA

Posteriormente se sacan las corrientes totales en cada uno de los modos, y sus respectivos tiempos, ver Tabla 14.

*Tabla 14.
Variables estrategia electrónica de control. Fuente Autor.*

Tiempos entre transmisiones		
Nombre	Intervalos (Minutos y segundos)	Intervalos (Horas)
Tiempo de consumo modo suspensión (Tcs)	0-10Min	0h - 0.166667 h
Tiempo de consumo modo transmisión 1 (Tct1)	10Min – 10Min 1s	0.166667 h - 0.166944778 h
Tiempo de consumo modo adquisición (Tca)	10Min 1s – 10Min 31s	0.166944778 h - 0.175278108 h
Tiempo de consumo modo transmisión 2 (Tct2)	10Min 31s – 10Min 32s	0.175278108 h - 0.175555886 h
Corrientes Totales		
Nombre	Consumo	
Consumo de corriente - modo suspensión (i1)	0.021mA	
Consumo de corriente - modo transmisión temperatura (i2)	195.2mA	
Consumo de corriente - modo normal-adquisición GPS (i3)	57.2mA	
Consumo de corriente - modo transmisión GPS (i4)	200.2mA	
Periodo (T)		
0.175555886 h		

Teniendo los valores totales de corriente y los tiempos entre transmisión se calcula la corriente media, lo que da como resultado el consumo de todo el sistema.

$$\text{Consumo(estrategia)} = \frac{1}{T} \left(\int_0^{T_{cs}} i_1(t) dt + \int_{T_{cs}}^{T_{ct1}} i_2(t) dt + \int_{T_{ct1}}^{T_{ca}} i_3(t) dt + \int_{T_{ca}}^{T_{ct2}} i_4(t) dt \right) \text{ (Ecuacion 7)}$$

En este punto se desarrollan las operaciones de cada una de las integrales, colocando la corriente y los intervalos de tiempo.

$$\text{Consumo(estrategia)} = \frac{1}{T} [(0.021mA)(0.166667h - 0h) + (195.2mA)(0.166944778h - 0.166667h) + (57.2mA)(0.175278108h - 0.166944778h) + (200.2mA)(0.17555886h - 0.175278108h)] \text{ (Ecuacion 8)}$$

$$\text{Consumo(estrategia)} = \frac{1}{T} [(0.021mA)(0.166667h) + (195.2mA)(0.000277778h) + (57.2mA)(0.00833333h) + (200.2mA)(0.000277778h)] \text{ (Ecuacion 9)}$$

$$\text{Consumo(estrategia)} = \frac{1}{0.17555886} [0.5899999042 \text{ mA}] \text{ (Ecuacion 10)}$$

$$\text{Consumo(estrategia)} = 3.36mA \text{ (Ecuacion 11)}$$

Ahora se calcula el tiempo de descarga de la batería con la siguiente fórmula

$$\text{Tiempo de duracion de la bateria(estrategia)} = \frac{\text{Carga electrica}}{\text{Consumo electrico del dispositivo}} \text{ (Ecuacion 12)}$$

$$\text{Tiempo de duracion de la bateria(estrategia)} = \frac{850mAh}{3.36mA} \text{ (Ecuacion 13)}$$

$$\text{Tiempo de duracion de la bateria(estrategia)} = 252.92 \text{ horas (Ecuacion 14)}$$

El tiempo de duración de la batería aplicando la estrategia de control para bajo consumo es de 10 días 12 horas 55 min y 12 seg, en comparación con el tiempo de descarga en modo normal de la ecuación 6, la diferencia es de 7 días 23 horas y 24 min, por lo que se puede evidenciar un ahorro en el consumo.

3.1.6 DISEÑO MECANICO

Para realizar la integración mecánica del prototipo se realizó un diseño 3D en el programa SOLID EDGE [66], donde se definieron las partes (caja, placa pcb y correa) que conforman el dispositivo, ver

Ilustración 15. A continuación, se describe cada parte:

1. Se muestra la cara frontal del prototipo donde se puede observar la placa pcb con sus respectivos componentes.

2. En la parte trasera de la caja se puede visualizar el sensor de temperatura, gracias a una perforación. Es necesario mencionar que el sensor se ubica en esa posición para facilitar la medición en el cuello del bovino.
3. Se presenta el diseño mecánico de la caja en conjunto con el collar y la placa pcb.

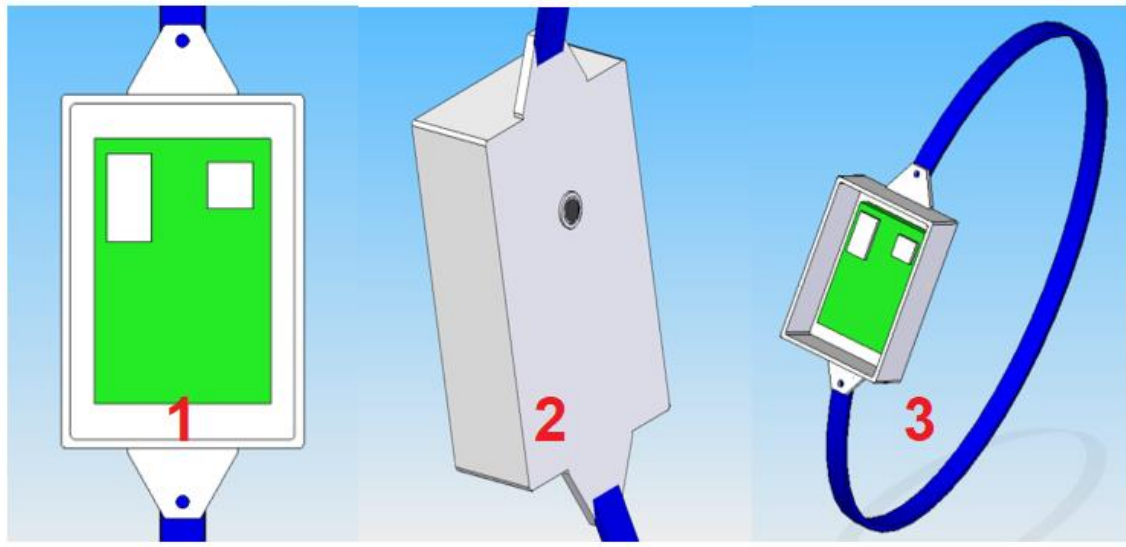


Ilustración 15. Diseño 3D Solid Edge. Fuente Autor.

Por otro lado, se realizó un diseño 3D de la placa electrónica, con sus respectivos componentes, para esto se utilizó la herramienta POV-Ray la cual permite generar imágenes foto realistas, ver Ilustración 16.

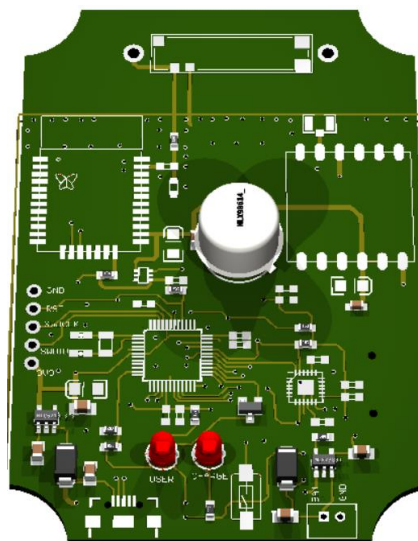


Ilustración 16. Diseño 3D PovRay. Fuente Autor.

Una vez realizado el diseño 3D de la tarjeta PCB, se procede a unir todas las piezas del prototipo, para esto se emplea una caja de plástico impermeable usada en proyectos electrónicos con dimensiones de 83x58x35mm.

Con el fin de instalar correctamente el sensor de temperatura dentro de la caja, se hace una abertura en la cara inferior. Por otro lado, la correa elegida es de un material llamado reata que resalta por su dureza y resistencia, ver Ilustración 17.



Ilustración 17. Diseño mecánico del sistema. Fuente Autor.

3.2 DISEÑO SOFTWARE DEL SISTEMA

Este capítulo expone lo relacionado al diseño firmware del sistema, en primer lugar, se realiza un diagrama de casos de uso en formato UML (Lenguaje unificado de modelado) que permite tener una visión general del funcionamiento de todo el sistema, ayudando así a capturar los requerimientos que el software debe cumplir [67]. Posteriormente, se realiza la elección de las herramientas software donde se presenta una tabla comparativa resaltando las características de cada tecnología, esto con el fin de elegir la que más se acomode a los requisitos establecidos en anteriores apartados. Finalmente se muestra la implementación de las plataformas seleccionadas.

3.2.1 DEFINICIÓN DE REQUERIMIENTOS FUNCIONALES, CASOS DE USO

Para que el sistema se ejecute adecuadamente y cumpla con todas las funcionalidades propuestas, es necesario definir los requerimientos funcionales y los casos de uso UML (Lenguaje unificado de modelado), el cual permite mostrar de forma gráfica el funcionamiento de un sistema cuando interactúa con él usuario u otros sistemas [67].

A continuación, se muestra la tabla de requerimientos. Ver Tabla 15.

*Tabla 15.
Requerimientos funcionales. Fuente Autor*

REQUERIMIENTOS FUNCIONALES	JUSTIFICACIÓN
1. El dispositivo debe enviar la información a internet y a su vez a una plataforma IoT.	Con el fin de poder visualizarla, analizarla para determinar datos relevantes de interés.
2. La frecuencia de envío de datos debe ser configurable.	Esto se hace a partir del uso de herramientas como el RTC que permitirán configurar el tiempo de adquisición y envío de la información suministrada por los sensores
3. Se debe poder monitorear la variable temperatura.	A través del uso de un sensor con buena precisión, de bajo consumo y que no sea invasivo
4. Se debe poder monitorear la variable ubicación.	Con el uso de un sensor gps que permitirá adquirir la latitud, longitud de forma precisa para así poder tener datos exactos, además, otras variables como velocidad y aceleración.
5. Es necesario que el sistema almacene los datos.	Esto con la utilización de las herramientas IoT que se encuentran en la red, se generan históricos e información estadística de gran interés
6. Generar alertas o avisos desde el firmware.	La lógica de programación que se realice podrá identificar los datos que exigen de mayor atención y es ahí donde se informará al usuario de que algo está pasando.
7. Poder tener un bajo consumo energético.	Al usar componentes que no exigen mucho energéticamente hablando y con la utilización de firmware que permita dormir o apagar distintos componentes del sistema cuando no se están usando.

3.2.1.1 DIAGRAMA CASOS DE USO

En la Ilustración 18, se muestra el diagrama de casos de uso, el cual consta de dos actores, usuario y sistema. El usuario puede ejecutar las acciones iniciar sesión e ingresar información, donde debe llenar los datos de ID y contraseña, en el momento que la información sea digitada, se hará el login en la plataforma IoT y posteriormente será almacenada. Por otro lado, la plataforma de programación se encarga de leer y enviar los datos de temperatura y ubicación a Sigfox, donde se realiza la recepción y codificación de la información recibida, ya en este punto la plataforma IoT es la encargada de decodificar y almacenar los datos. Finalmente, el usuario tendrá acceso a los valores de temperatura, ubicación y el id de usuario proporcionados por la plataforma IoT.

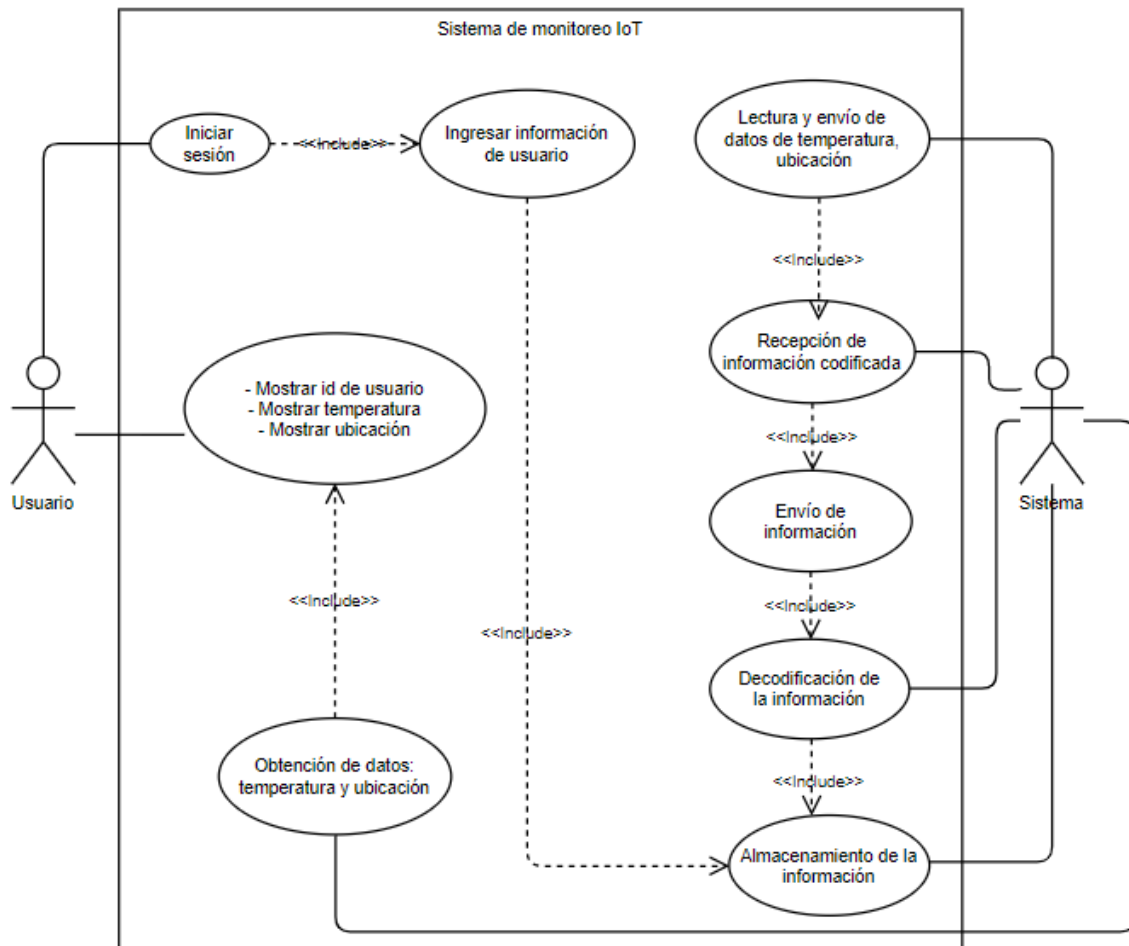


Ilustración 18. Diagrama casos de uso UML. Fuente Autor.

3.2.2 SELECCIÓN DE HERRAMIENTAS DE DESARROLLO

El presente capítulo está dividido en dos etapas, la primera de ellas es la selección de herramientas para el entorno de desarrollo, que comprende toda la parte lógica de programación que será utilizada para el correcto funcionamiento del sistema y la segunda es la selección de la plataforma IoT, donde se visualizarán los datos enviados desde Sigfox y posteriormente serán analizados.

3.2.2.1 SELECCIÓN DEL ENTORNO DE DESARROLLO Y PROGRAMACIÓN

La plataforma IDE de Arduino ha sido elegida como el entorno de desarrollo por los siguientes factores, el primero es que el microcontrolador elegido para el desarrollo del presente proyecto se puede programar bajo esta plataforma, otro de los motivos fundamentales para la elección de este entorno es la experiencia con la que se cuenta para manejar la herramienta. Por último, pero no menos importante son las características que posee (sistema de código abierto, fácil comprensión, soporte global, variedad de librerías), las cuales hacen de Arduino IDE un entorno muy utilizado [68].

3.2.2.2 SELECCIÓN PLATAFORMA IOT





En los sistemas IoT existen generalmente dos servicios para el tratamiento de los datos, en primer lugar, está el servidor de datos que conforma lo que es el backend de Sigfox, el cual se encarga de la recepción y asegura que los datos estén disponibles para ser tratados. Además esta herramienta cuenta con un servicio de computación en la nube que permite redirigir la información a cualquier aplicación de usuario, donde se podrá visualizar para su posterior análisis [59].

En segundo lugar, están los llamados servicios de aplicación, en los que se implementa procesamiento de datos y visualización.

Para la elección del servicio de cloud computing se deben tener en cuenta características tales como: que sea compatible con el backend de Sigfox, que permita la decodificación de la información, que cuente con las opciones de creación de tableros y alertas, entre otras.

En la Tabla 16 se muestra la comparación de diferentes plataformas IoT que cumplen con los requerimientos del proyecto.

Tabla 16.
Comparación de plataformas IoT. Fuente [69], [70], [71], [72].

Plataformas IOT	Ubidots 	Thingspeak 	Microsoft Azure IoT HUB 	Amazon AWS 
Conectividad	-Sigfox -Wi-Fi -LoRaWAN -LTE-M -NB-IoT -Ethernet -Celular	-LoRaWAN -Libelium -Beckhoff -Particle	-Sigfox -Zigbee -Z-Wave -LoRaWAN -Wifi	-Sigfox -LoRaWAN -NB-IoT -Wi-Fi -LTE-M
Hardware	-Arduino -Espressif system -Microchip -Particle -Intel -Raspberry pi -Mc things -Siemens -Stm	-Electronic imp -Arduino -Raspberry Pi	-Arduino -Esp8266 -Raspberry Pi	-ESP8266 -Arduino -Raspberry Pi
Lenguajes soportados	Python, Java, C, PHP, Node.js, Ruby	Matlab	-C, C++,C# -Python -Node.js -Java -PHP -Go	-Ruby -Python -C++ -Node.js -Java -Go

			-.NET	-.NET -PHP
Protocolos	-HTTP -MQTT -TCP -UDP	-MQTT -HTTP	-MQTT -HTTP -AMQP	-MQTT -HTTP -WebSockets
Ventajas	-Fiabilidad global -Interfaces intuitivas -Dispone de numerosas librerías. -Posee amplia sección de tutoriales y SDK's -De uso simple y claro -Prueba gratuita por un mes.	-La plataforma es de código abierto. -Fácil de utilizar	-Gran variedad de lenguajes de programación -Seguridad entre las conexiones IoT	-Libertad de lenguaje de programación -Integración con todos los servicios ofrecidos por Amazon aws. -Análisis de la información.
Desventajas	-Los precios por paquete son elevados	-La documentación es limitada frente a algunos dispositivos de hardware	-Complejidad en la configuración de los dispositivos IoT a la plataforma -Precios elevados para subir los datos	-Plataforma compleja para principiantes --Precios elevados para subir los datos

Se optó por trabajar con un sistema de cloud computing, específicamente Ubidots, debido a que tiene bloques funcionales integrados, los cuales se pueden configurar acorde a las necesidades requeridas, facilitando así el desarrollo del proyecto. Esta es una plataforma asequible, confiable y de las más utilizadas en aplicaciones IoT [69].

Ubidots posee diferentes aspectos a destacar: facilidad de uso ya que es intuitiva, rápida al usuario y ofrece distintas formas para el análisis de datos. Así mismo la plataforma maneja estándares de protocolos que permiten el empleo de datos provenientes de diferentes tecnologías como es el caso Sigfox, LoRaWAN, Wi-Fi, y demás, por otro lado, soporta distintos lenguajes de programación y protocolos de comunicación que hacen más fácil el desarrollo de un proyecto, y por último su amplia comunidad que tiene como propósito ayudar, sostener y hacer crecer la tecnología IoT [69].

3.2.3 IMPLEMENTACIÓN DE HERRAMIENTAS SOFTWARE

A continuación, se presenta una descripción del funcionamiento de las diferentes herramientas software utilizadas dentro del proyecto, ver Ilustración 19.

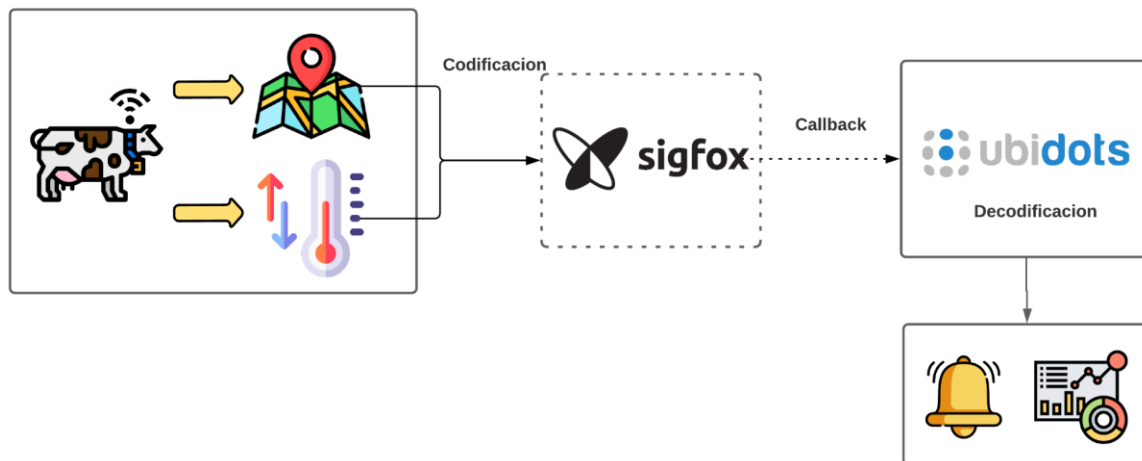


Ilustración 19. Diagrama de implementación software. Fuente Autor.

3.2.3.1 DESARROLLO DE FIRMWARE CON LA HERRAMIENTA ARDUINO IDE

Con esta herramienta se realizó la adquisición y el envío de datos a la nube de Sigfox, en primer lugar, se recolectan las señales (Temperatura, GPS), y posteriormente se procede a codificar la información.

Es importante recordar que los mensajes enviados al backend de Sigfox tienen un límite en su tamaño, esto depende del tipo (ascendente o descendente), para este caso son ascendentes, por lo que su tamaño es de 12 Bytes. En la Ilustración 20, se muestra el formato de la información enviada, donde el payload representa los datos medidos por los sensores.

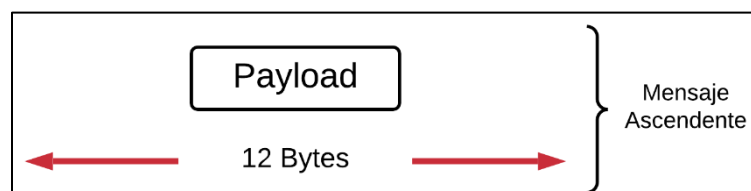


Ilustración 20. Estructura de mensaje Sigfox. Fuente Autor.

Teniendo en cuenta que los mensajes tienen un límite en su tamaño se debe optar por codificar los datos de tal forma que se aproveche el espacio al máximo y no se pierda precisión, es por esta razón que se hará uso de la norma IEEE 754 que permite convertir los números que están en formato flotante a hexadecimal

Para realizar la conversión de formato flotante a hexadecimal con el estándar IEEE 754 el cual se divide en precisión simple y precisión doble, la precisión simple cuenta con las siguientes características [73].

- Signo: Utiliza un bit. Si el número es negativo se representa con 1 y si por el contrario es positivo con 0.
- Exponente: Utiliza 8 bits y representa exponentes negativos o positivos
- Mantisa: Consta de 23 bits y muestra los dígitos significativos de un numero en forma binaria

En la Ilustración 21 se muestra la estructura de la norma IEEE 754, con todas las partes anteriormente mencionadas.

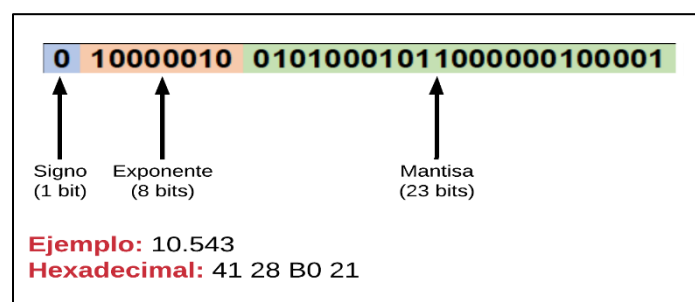


Ilustración 21. Formato de punto flotante IEEE-754. Fuente[74].

Una vez que los datos se pasen a hexadecimal con el protocolo anteriormente descrito, se obtendrán 2 tramas, una de GPS y la otra de temperatura, cada una de ellas será nombrada con los identificadores 01 y 02 respectivamente.

Las tramas de temperatura y GPS quedarán compuestas de la siguiente forma, ver Ilustración 22.

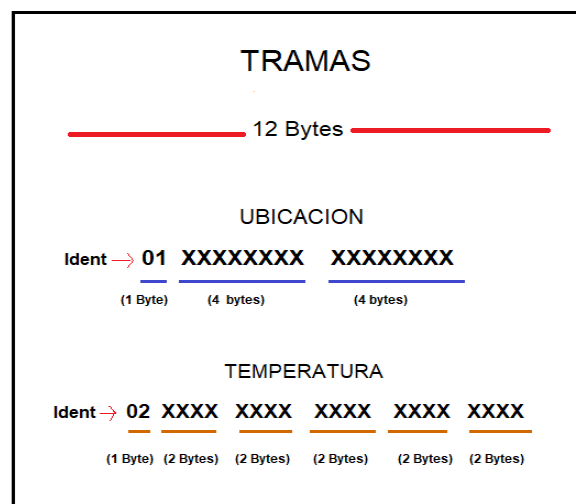


Ilustración 22. Tramas de envío. Fuente Autor.

En la Tabla 17 se presenta más en detalle todo el proceso de codificación para las variables (Temperatura y GPS), con la norma IEEE 754 y posteriormente la conversión a hexadecimal.

*Tabla 17.
Tramas Sigfox. Fuente Autor.*

Payload (12 Bytes)						
Ubicación	Identificador	Latitud			Longitud	
	01	-21.55667			108.34333	
	01 (1 Byte)	C1AC740F (4 Bytes)			42D8AFC9 (4 Bytes)	
Temperatura	Identificador	Tem1	Tem2	Tem3	Tem4	Tem5
	02	20.32	21.56	20.10	19.80	21.14
	02 (1 Byte)	41A2 (2 Bytes)	41AC (2 Bytes)	41A0 (2 Bytes)	419E (2 Bytes)	41A9 (2 Bytes)

Ya teniendo la información codificada de forma correcta se procede a utilizar los comandos AT que controlan el transceptor Wisol y permiten el envío de datos a Sigfox, a continuación, en la Tabla 18 se describen los más relevantes para el proyecto.

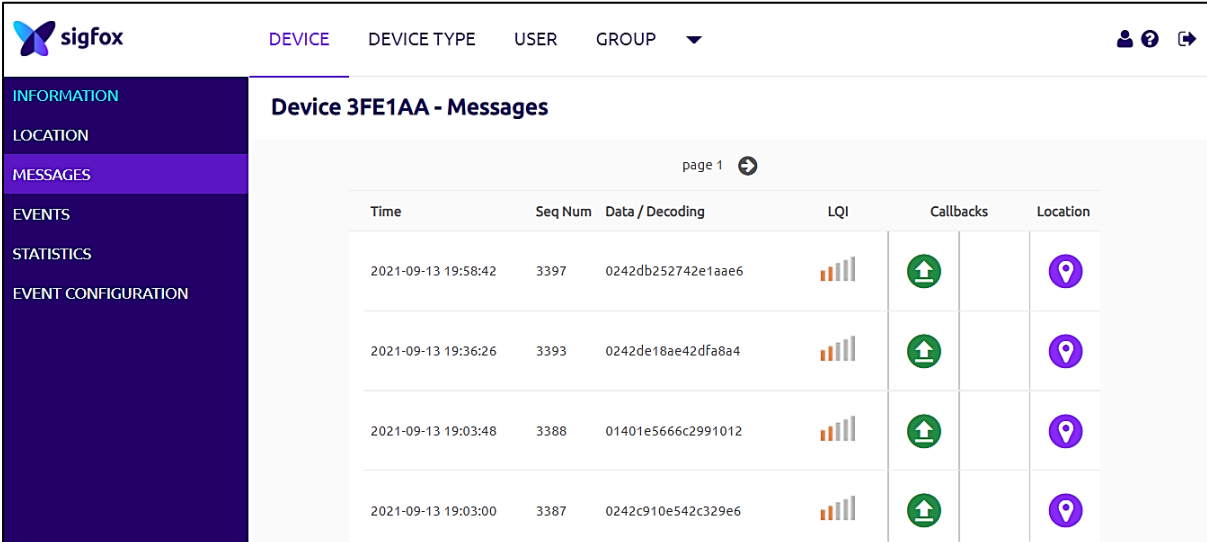
*Tabla 18.
Comandos AT-Modulo Wisol. Fuente [75].*

Comando	Nombre	Descripción
AT	Comando de prueba	Devuelve 'OK'. Puede utilizarse para comprobar la comunicación
AT\$I=10	Información	Obtener el ID del dispositivo
AT\$I=11	Información	Obtener el PAC del dispositivo
AT\$IF=920800000	Ajustar la frecuencia de transmisión	Onda de modulación encendida para el RCZ4
AT\$CB=-1,1	Modo de prueba: Byte constante de transmisión	Se usa para las pruebas de emisión, es útil enviar un patrón de bits específico.
AT\$RC	Restablecer el canal por defecto	Se utiliza para restablecer el dispositivo en el canal de macro SIGFOX predeterminado.
AT\$SF	Enviar trama	Enviar datos de carga útil, de 1 a 12 bytes.

3.2.3.2 MANEJO DEL BACKEND DE SIGFOX

El backend de Sigfox recibe los datos provenientes del módulo Wisol, y posteriormente permite visualizarlos, además muestra información relevante como fecha, hora, intensidad de envío y recepción, entre las más destacadas, ver Ilustración 23.

Sigfox cuenta con una herramienta denominada callback que permite la conexión entre el backend y un servidor externo [59]. En el anexo A se presentan los pasos a seguir para enviar datos a Ubidots mediante la creación de un callback.



The screenshot displays the Sigfox web interface for a specific device. The left sidebar contains navigation options: INFORMATION, LOCATION, MESSAGES (highlighted), EVENTS, STATISTICS, and EVENT CONFIGURATION. The main content area is titled 'Device 3FE1AA - Messages' and shows a table of message logs. The table has six columns: Time, Seq Num, Data / Decoding, LQI, Callbacks, and Location. Each row represents a message received from the device, with a corresponding LQI bar chart, a green callback icon, and a location pin icon.

Time	Seq Num	Data / Decoding	LQI	Callbacks	Location
2021-09-13 19:58:42	3397	0242db252742e1aae6			
2021-09-13 19:36:26	3393	0242de18ae42dfa8a4			
2021-09-13 19:03:48	3388	01401e5666c2991012			
2021-09-13 19:03:00	3387	0242c910e542c329e6			

Ilustración 23. Backend de Sigfox. Fuente Autor.

3.2.3.3 MANEJO DE LA PLATAFORMA UBIDOTS

Como se mencionó en anteriores capítulos, Ubidots es un servicio de aplicación escogido para almacenar y analizar la información que llega desde el backend de Sigfox, para realizar el envío de datos es necesaria la creación de un callback, esta herramienta permite realizar la conexión entre el backend de Sigfox y la plataforma Ubidots.

Una vez enviados los datos desde el backend de Sigfox, se debe hacer uso de la herramienta UbiFunctions que permite decodificar los datos de Ubicación y temperatura, luego se crea un dashboard donde se obtiene visualmente la información por medio de widgets dinámicos (Ver Ilustración 24) [76], en ellos se pueden adquirir historiales, gráficos y demás, lo cual facilita tener un control del proceso o actividad que se está monitoreando. Los pasos para llevar a cabo la configuración de la plataforma de Ubidots se encuentran en el Anexo B.

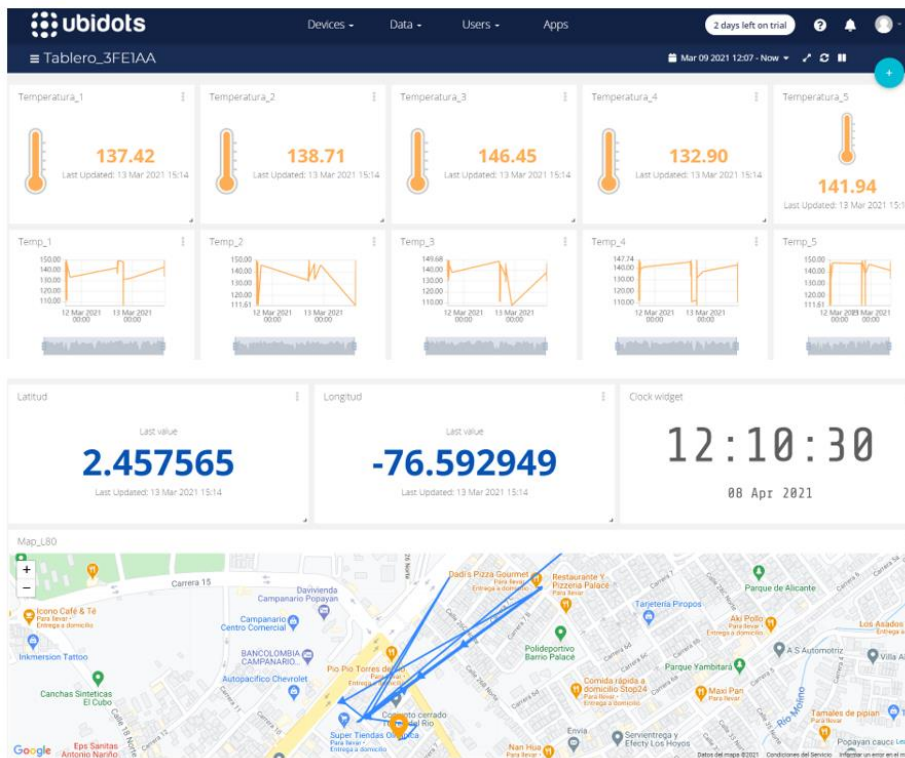


Ilustración 24. Dashboard Ubidots. Fuente [69].

Por otro lado, Ubidots cuenta con la opción eventos, que permite configurar diferentes tipos de alertas, por valor, por posición o por estado de actividad, en la Ilustración 25 se muestra un evento por posición, es decir geocerca. Si la alerta se cumple según una configuración se envía una notificación que puede ser por mensaje de texto, correo electrónico y demás.

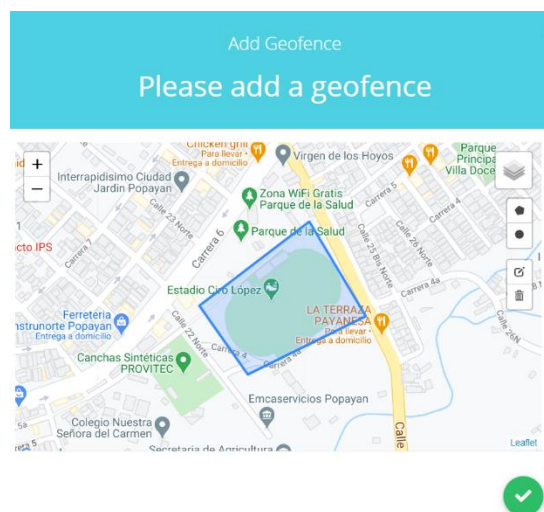


Ilustración 25. Geocerca Ubidots. Fuente [69]

3.3 IMPLEMENTACIÓN DE SISTEMA ELECTRÓNICO

En primer lugar, el sistema electrónico inició con un prototipo el cual constaba de un módulo ESP32 con el fin de poder hacer la comunicación con Ubidots, se hizo la lectura analógica de un potenciómetro, Ver Ilustración 26.

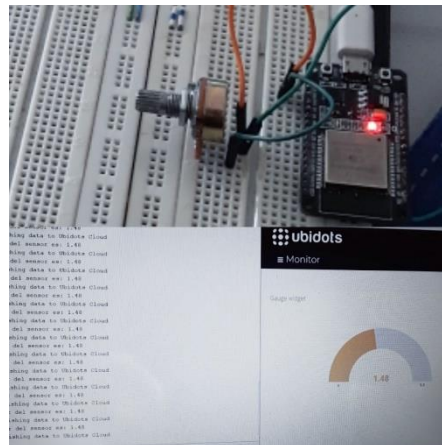


Ilustración 26. Envío de datos con ESP32. Fuente autor.

En la siguiente etapa se integró el módulo Wisol de Sigfox y el microcontrolador ATSAM21 en la misma placa PCB, con el fin de hacer el envío de datos al backend de Sigfox, ver Ilustración 27 Ilustración 27.

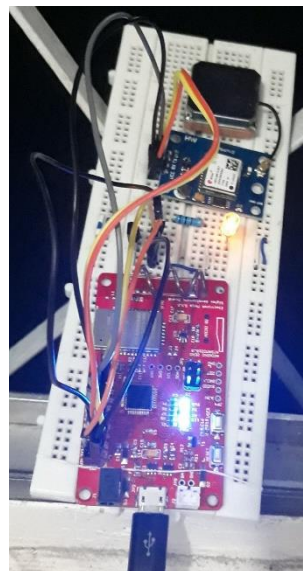


Ilustración 27. Placa pcb con módulo Wisol y ATSAM21. Fuente Autor.

Una vez integrado el módulo Wisol junto con el microcontrolador SAMD21 en la misma tarjeta se procede a incorporar el módulo GPS L80-R, ver Ilustración 28.

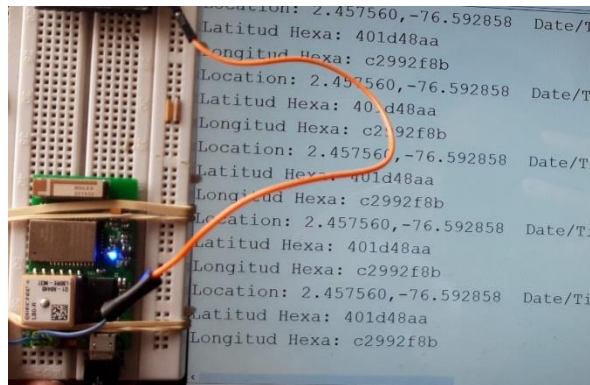


Ilustración 28. Placa pcb con módulo Wisol, ATSAMD21 y L80-R. Fuente Autor.

3.3.1 PROTOTIPADO ELECTRÓNICO – PRUEBAS PARCIALES

En este apartado se presenta el procedimiento para verificar el correcto funcionamiento de cada uno de los sensores (temperatura, GPS). Cabe resaltar que en posteriores capítulos se comprobará el funcionamiento del prototipo ya integrado con todos sus componentes.

Prueba sensor de temperatura MLX90614

Para realizar las mediciones con el sensor MLX90614 se deben seguir los siguientes pasos, primeramente, se reconoce cada uno de los pines con la función que cumplen en el sensor, teniendo en cuenta esta información se realiza el montaje con sus correspondientes conexiones, ver Ilustración 29.

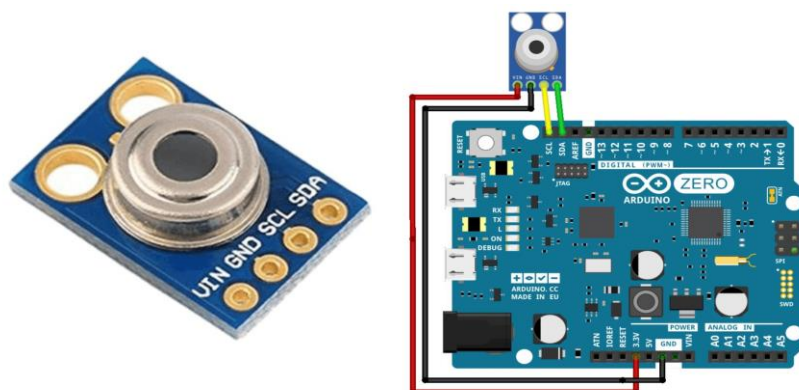


Ilustración 29. Sensor de temperatura MLX90614. Fuente [77].

Seguidamente de las conexiones, se debe desarrollar un código en Arduino IDE, que permita leer y mostrar los datos que arroja cada dispositivo conectado a la placa

arduino, a continuación, se muestra un diagrama de flujo que describe la lógica que se utilizó para obtener los valores de temperatura, ver Ilustración 30.

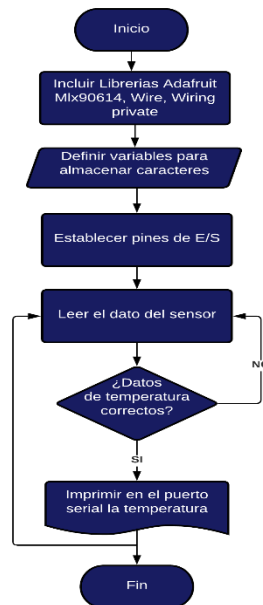


Ilustración 30. Diagrama de flujo MLX90614. Fuente Autor.

Finalmente, para comprobar que los datos enviados al monitor serial son correctos, se hace uso de un termómetro digital certificado de la marca Elechico. En la siguiente tabla se muestra la comparación entre ambas mediciones, ver Tabla 19.

*Tabla 19.
Comparación de medición de temperatura entre CCIR1001 Y LMX90614. Fuente Autor.*

NUMERO	TERMOMETRO DIGITAL (CCIR1001) (°C)	MLX90614 (°C)	ERROR ABSOLUTO (°C)
1	34.6	34.55	0.05
2	34.1	34.00	0.1
3	34.4	34.39	0.01
4	34.7	34.61	0.09
5	35.2	35.16	0.04
6	34.8	34.71	0.09
7	35.4	35.28	0.12
8	35.1	35.07	0.03
9	35.2	35.25	0.05
10	34.6	34.70	0.1

En la Ilustración 31 se muestran las pruebas realizadas con el sensor MLX90614 y él termómetro digital de Elechico.

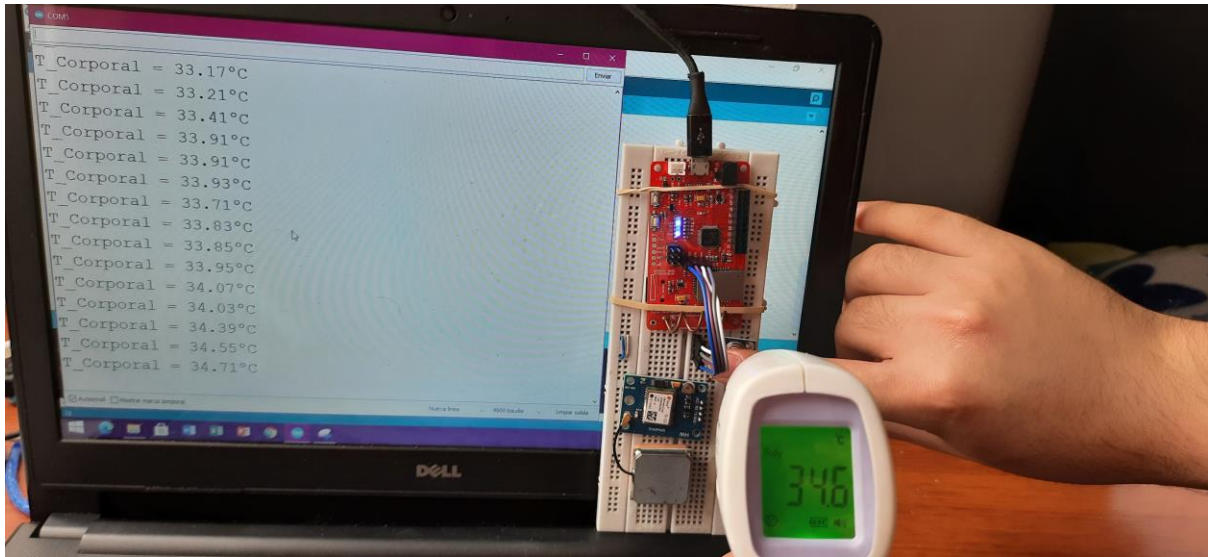


Ilustración 31. Prueba medición temperatura MLX90614 y CCIR1001. Fuente Autor.

Prueba módulo GPS L80-R

En primer lugar, se reconoce cada uno de los pines con la función que cumplen en el sensor, luego se procede a hacer el montaje del módulo con sus respectivas conexiones ver, Ilustración 32 Ilustración 32.

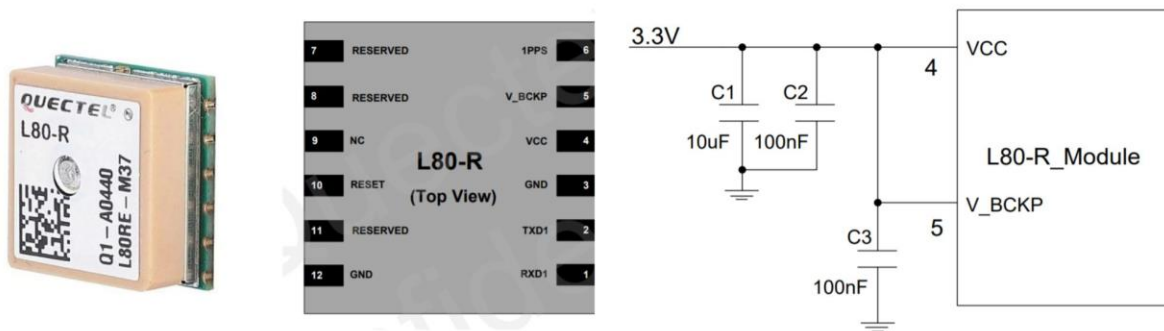


Ilustración 32. GPS L80-R. Fuente [38], [78].

Con el fin de poder observar los datos que proporciona el GPS L80-R, se procede a realizar un código en Arduino IDE usando librerías para la lectura del dispositivo. En la Ilustración 33 se muestra un diagrama de flujo que describe la lógica general que se utilizó para obtener los valores de ubicación.

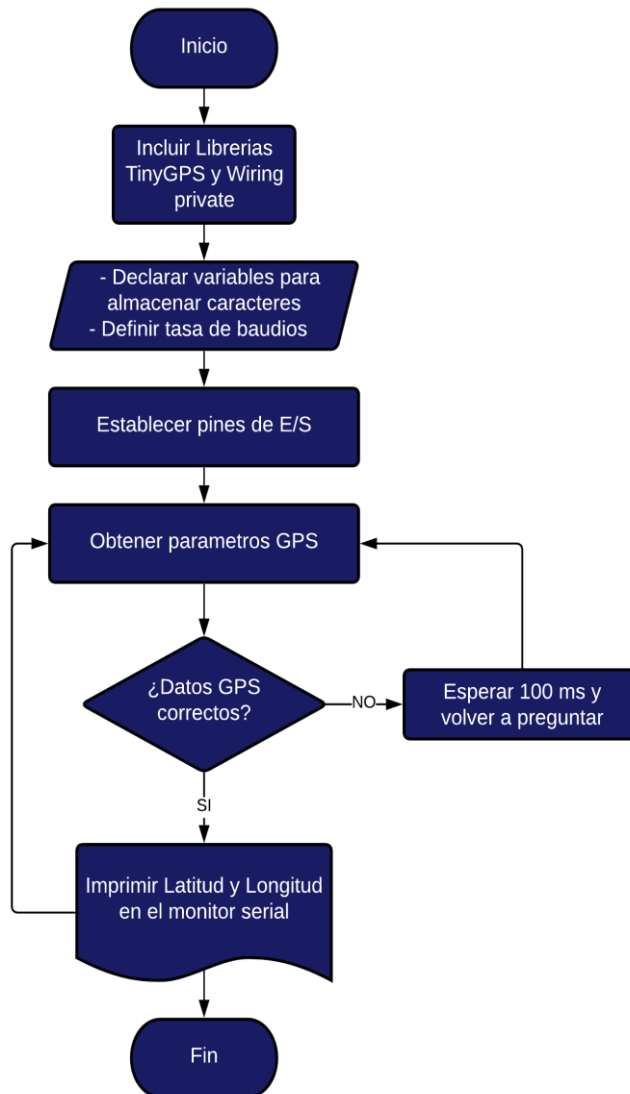


Ilustración 33. Diagrama de flujo L80-R. Fuente Autor.

Por último, para comprobar que los datos enviados del módulo L80-R al monitor serial son correctos, se hace uso de la herramienta Google Maps. En la siguiente tabla se muestra la comparación de ambas mediciones, ver Tabla 20.

Tabla 20.
Comparación de medición de ubicación GPS, entre L80-R Y Google Maps. Fuente Autor.

NUMERO	GOOGLE MAPS		L80-R		ERRORES ABSOLUTO (°C)	
	LAT	LON	LAT	LON	LAT	LON
1	2.457575	- 76.592903	2.457600	- 76.592828	0.000025	0.000075
2	2.457660	- 76.592859	2.457605	- 76.592820	0.000055	0.000039
3	2.457753	- 76.592736	2.457827	- 76.592707	0.000074	0.000029
4	2.457964	- 76.592787	2.458034	- 76.592857	0.00007	0.00007
5	2.457888	- 76.593124	2.457794	- 76.593166	0.000094	0.000042
6	2.457673	- 76.592771	2.457679	- 76.592736	0.000006	0.000035
7	2.457453	- 76.593106	2.457328	- 76.593120	0.000125	0.000014
8	2.457532	- 76.593190	2.457642	- 76.593222	0.000110	0.000032
9	2.457845	- 76.593164	2.457999	- 76.593344	0.000154	0.00018
10	2.457978	- 76.593091	2.458135	- 76.592885	0.000157	0.000206

En la Ilustración 34 se muestran las mediciones del gps L80-R y la herramienta google maps

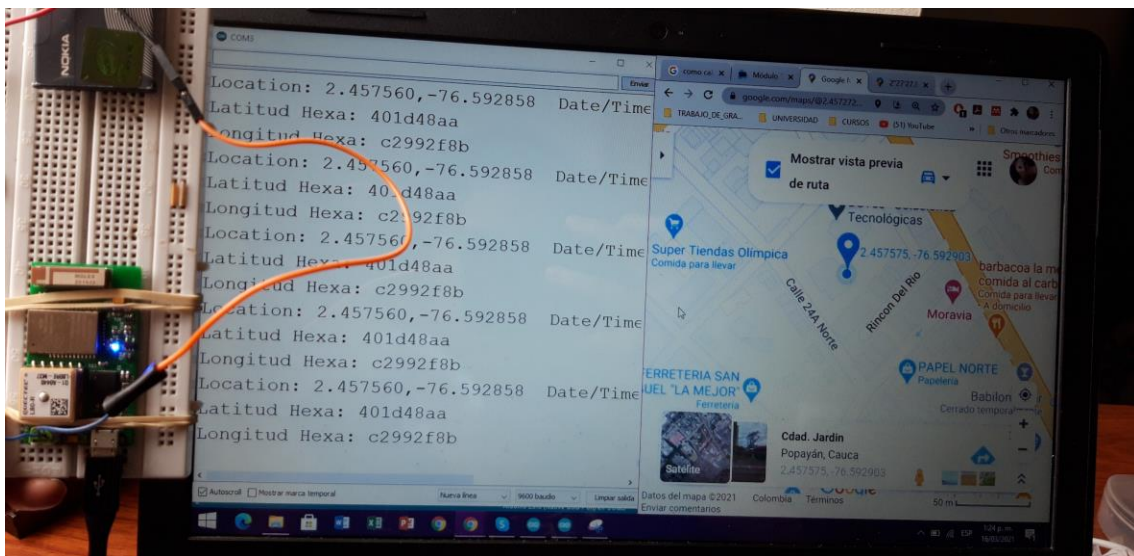


Ilustración 34. Prueba medición de la ubicación entre L80-R Y Google Maps. Fuente Autor

3.3.2 PRUEBAS DE LABORATORIO DE SISTEMA ELECTRÓNICO – CIRCUITO INTEGRADO

En este apartado se hacen las pruebas del sistema con todos sus componentes integrados. Inicialmente se realizaron mediciones con los sensores (temperatura, GPS) y se verifica que los datos obtenidos sean correctos, posterior a esto se envió la información codificada al backend de Sigfox a través del módulo Wisol, esto con el fin de comprobar que el transceptor está respondiendo a los comandos AT enviados desde el microcontrolador SAMD21.

Una vez comprobada la comunicación entre el módulo Wisol y el backend de Sigfox, se procede a enviar la información a Ubidots por medio de un callback, donde los datos serán decodificados, y analizados.

A continuación, en la Ilustración 35 se muestran los datos que llegan a la plataforma de Ubidots.

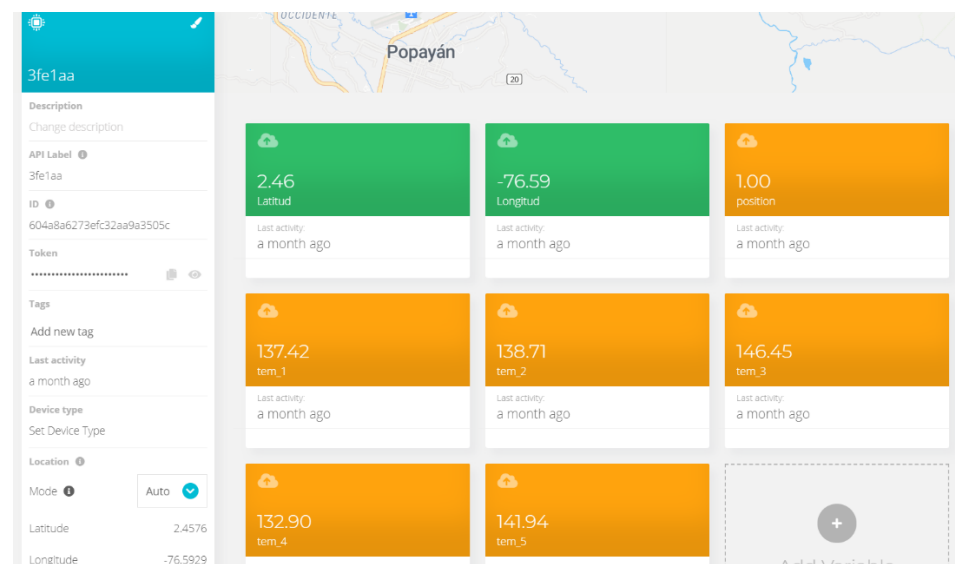


Ilustración 35. Tablero principal Ubidots. Fuente Autor.

4. EVALUACIÓN DEL SISTEMA ELECTRÓNICO

Con la finalidad de evaluar el sistema desarrollado, se presenta un protocolo de pruebas, el cual se debe seguir con el fin de poner en funcionamiento el prototipo bajo los parámetros establecidos. Finalmente, con las pruebas realizadas y los datos obtenidos se sacan los resultados.

4.1 DISEÑO DE PROTOCOLO DE PRUEBAS

Al realizar este protocolo de pruebas se tiene como objetivo comprobar el funcionamiento del sistema en un entorno real y así obtener datos para caracterizar la variable de temperatura.

Con la aplicación de este protocolo se espera contar con la mayor cantidad de datos posibles a fin de establecer un análisis de correlación entre las variables y el evento de estro en el bovino.

A continuación, se listan los materiales requeridos para llevar a cabo el protocolo de pruebas.

- Tarjeta del prototipo (Acondicionada con su collar).
- Suscripción activa del servicio de comunicación de datos de la red de Sigfox.
- Suscripción activa del servicio de Ubidots.
- Computador portátil con acceso a internet.
- Cable de datos micro USB
- Multímetro

A continuación, se sugieren los siguientes pasos para realizar las pruebas de adquisición de datos:

1. Se establece la hora inicial y final de la prueba, con el fin de llevar un registro de los datos para luego ser analizados junto con las variables medidas.
2. Verificar cobertura de la tecnología Sigfox en el lugar donde se va a realizar la prueba, esto debido a que las antenas de comunicaciones se encuentran distribuidas en puntos específicos.
3. Se elige el bovino (Hembra) que esté próximo a presentar características de estro.
4. Verificar el correcto funcionamiento del sistema (batería, comunicación Sigfox, Ubidots, GPS, sensor de temperatura) en laboratorio, previo a la instalación en él bovino con el fin de corregir posibles fallas
5. Instalar el prototipo en el cuello del animal elegido, mediante el uso de un collar.
6. Comprobar que los datos de temperatura y ubicación se están recolectando y enviando a la plataforma Ubidots.
7. Durante el tiempo de captura de datos se propone realizar inspecciones visuales periódicamente cada 2 días.

4.2 RESULTADOS OBTENIDOS DEL SISTEMA ELECTRÓNICO

En este capítulo se presentan los resultados obtenidos de cada una de las pruebas realizadas con el prototipo desarrollado, esto con la ayuda de imágenes que servirán para representar y corroborar la información obtenida.

- Los diagramas presentados en el apartado de diseño del plano esquemático ver Ilustración 12 y Ilustración 13, se realizaron teniendo en cuenta la hoja de datos de cada uno de los componentes y los requerimientos electrónicos establecidos en el proyecto, lo que resultó útil para la organización, y correcto funcionamiento del PCB, ver Ilustración 14. Sin embargo, en esta primera versión se presentaron 2 inconvenientes a la hora del ensamblaje, ambos problemas fueron de conexión con los dispositivos GPS L80-R y el sensor de temperatura MLX90614, ver Ilustración 36.

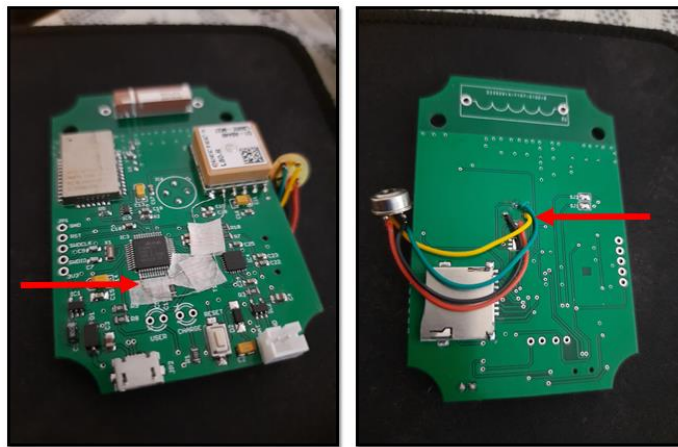


Ilustración 36. Primera versión PCB ensamblado. Fuente Autor.

- La segunda versión del PCB cuenta con todas las conexiones corregidas y fue evaluada en las pruebas de laboratorio. A pesar de ser más completa que la primera versión se notó que el módulo de comunicación Sigfox necesitaba una conexión adicional dentro del PCB, para mejorar su rendimiento en el envío de datos, ver Ilustración 37.

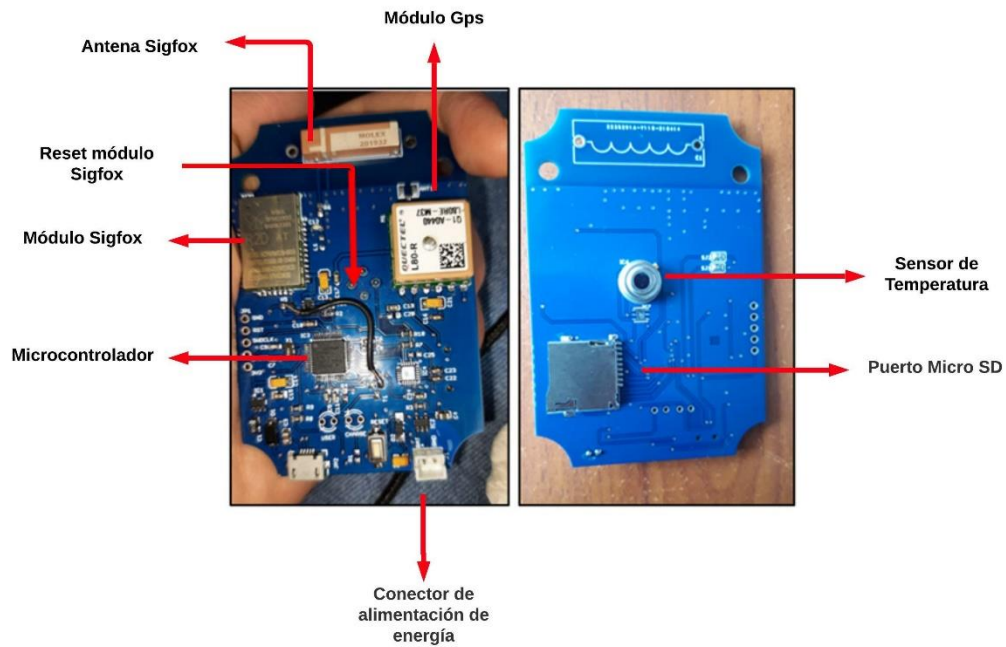


Ilustración 37. Segunda versión PCB ensamblado. Fuente Autor.

- En cuanto al diseño mecánico del prototipo fue necesario realizar cambios a la primera versión puesto que antes de colocarlo en el bovino, los ganaderos sugirieron que se pinte todo el collar de negro, esto con el fin de que pase desapercibido, y no sea manipulado por personas curiosas. Ver Ilustración 38.



Ilustración 38. Primera versión collar. Fuente Autor.

- Después de realizar algunas pruebas en campo se presentaron problemas con el funcionamiento del dispositivo, puesto que en los primeros días de pruebas el clima fue de invierno, ocasionando filtraciones de agua dentro de la caja. Otro de los inconvenientes presentados fue la inestabilidad del collar en la

vaca, ya que al tomar agua o pastar el collar se movía de dirección, ver Ilustración 39.

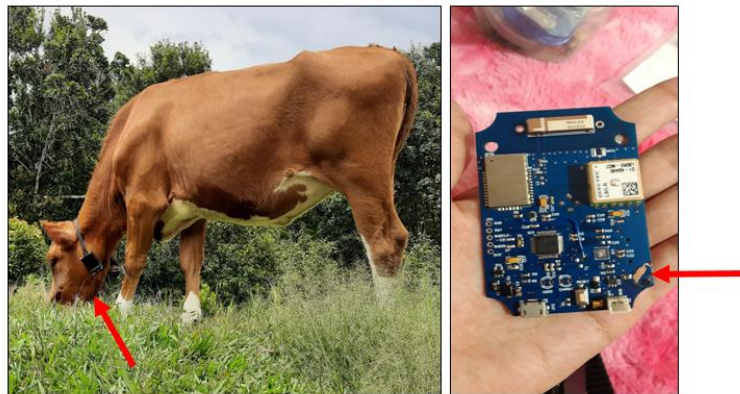


Ilustración 39. Problemas primera versión collar. Fuente Autor.

- La primera versión del collar, permitió identificar todos los problemas que se pueden presentar en las pruebas de campo, facilitando así una mejora en cada una de las partes que componen el prototipo. En primer lugar, se adquirió una nueva caja con las perforaciones exactas para el sensor de temperatura, además se obtuvieron empaques para la tapa de la caja y el orificio donde va el sensor MLX90614, con el fin de evitar filtraciones de agua dentro de la caja. Por otro lado se diseñó un cabezal con la ayuda del ganadero para mejorar la fijación del collar. Además, se colocó un contrapeso en la reata, logrando así que el prototipo no se deslice por el cuello de la vaca. Ver Ilustración 40.

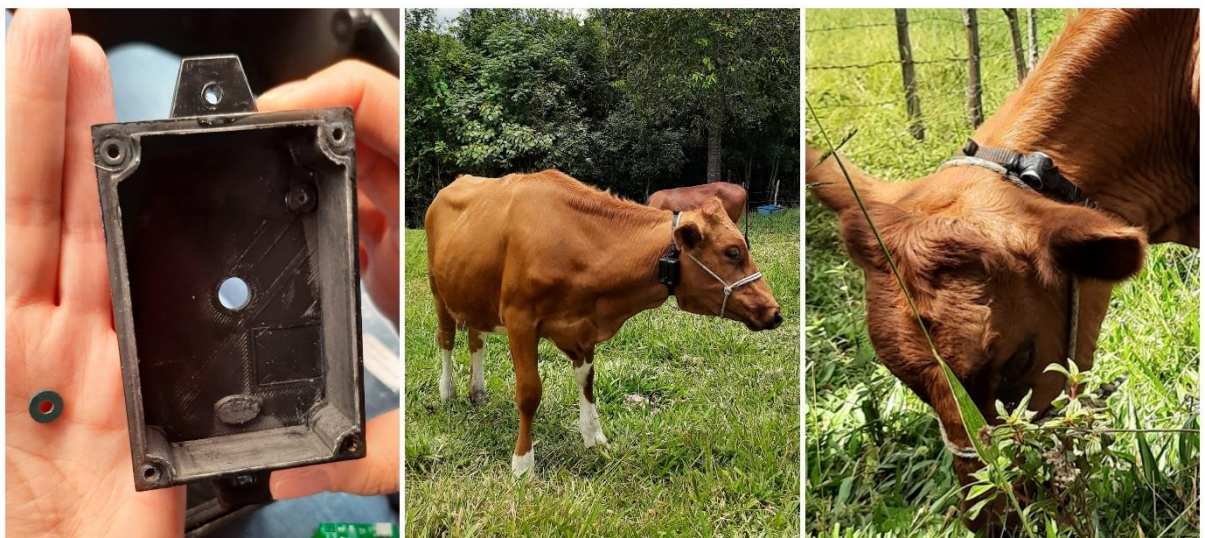


Ilustración 40. Ajustes collar. Fuente Autor.

- En la Ilustración 41 se puede observar como el día 11 de septiembre la temperatura corporal y ambiental decrecieron hasta 7.66°C y 13.31°C respectivamente. Una temperatura tan baja como esta representa que el

animal está muerto o ya no cuenta con el collar, efectivamente se corrobora al día siguiente que el bovino no tiene el collar puesto.



Ilustración 41. Grafica de datos 11/09/21. Fuente Autor.

- En la Ilustración 42, se puede apreciar como la temperatura ambiente, presenta unos picos que llegan hasta 38 °C aproximadamente, este error está relacionado a los movimientos bruscos del animal, y el calor que se encierra dentro de la caja.

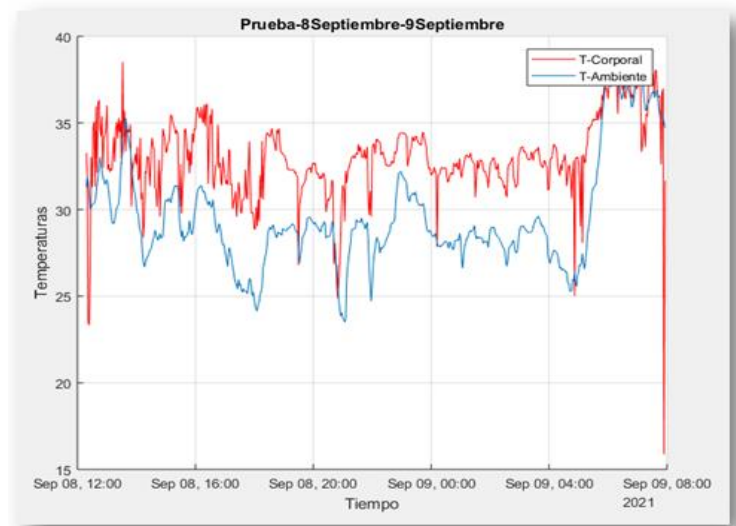


Ilustración 42. Grafica Temperaturas 8/09/21. Fuente Autor.

5. ANÁLISIS DE DATOS

Este capítulo describe el procedimiento que se llevó a cabo para el análisis de la variable temperatura. Comenzando con la adquisición de los datos, es decir cómo se obtuvieron en primera instancia, seguidamente de esta fase se realiza una segmentación que permite analizar información a mayor detalle, posteriormente se aplican diferentes pruebas estadísticas para el análisis, se muestran los resultados, y por último se plantea el algoritmo para la estimación de estado de estro.

5.1 ADQUISICIÓN Y TRANSFORMACIÓN DE LOS DATOS

Inicialmente la información es capturada por el dispositivo electrónico y almacenada por una tarjeta micro SD en un archivo “.CSV”, estos datos se transforman a filas y columnas con la ayuda del software Excel, ver Ilustración 43.

Es importante mencionar que se tomaron alrededor de 7372 muestras, durante 3 semanas aproximadamente.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	Column1	Column2	Column3	Column4	Column5	Column6	Column7	Column8	Column9	Column10	Column11	Column12	Column13	Column14
2	Fecha	Tiempo(ms)	TemC1	TemA1	TemC2	TemA2	TemC3	TemA3	TemC4	TemA4	TemC5	TemA5	Latitud	Longitud
3	12:7:21	0:2:3	33.33	31.87	31.07	32.01	31.83	31.91	32.61	31.87	31.01	31.73	2.472967	-76.532435
4	12:7:21	0:13:11	34.07	31.05	35.23	31.55	35.85	32.47	36.21	33.05	36.25	33.49	2.472957	-76.532427
5	12:7:21	0:24:19	35.99	33.71	36.17	33.73	31.99	33.65	32.47	32.93	33.27	32.43	2.472957	-76.532427
6	12:7:21	0:35:27	35.55	32.51	32.55	32.63	35.23	32.63	34.77	32.65	35.55	32.59	2.472957	-76.532427
7	12:7:21	0:46:35	36.15	32.67	34.33	32.67	32.81	32.81	35.97	32.59	36.15	32.85	2.472957	-76.532427
8	12:7:21	0:57:43	34.33	32.95	33.91	32.67	34.77	32.23	36.15	32.41	35.35	32.75	2.472957	-76.532427
9	12:7:21	1:8:51	35.19	33.11	32.77	32.81	34.75	32.49	32.25	32.25	35.19	32.33	2.472957	-76.532427
10	12:7:21	1:19:59	32.39	33.03	34.39	33.25	35.23	33.67	36.33	33.83	36.41	33.83	2.472957	-76.532427
11	12:7:21	1:31:7	37.33	34.83	35.85	35.53	36.31	36.03	35.31	36.09	36.05	35.51	2.472957	-76.532427
12	12:7:21	1:42:15	36.31	35.59	31.91	35.43	35.97	35.09	34.59	34.53	33.19	34.45	2.472957	-76.532427
13	12:7:21	1:53:23	35.77	34.79	30.51	34.65	36.21	34.75	36.71	34.91	31.19	34.77	2.472957	-76.532427

Ilustración 43. Datos almacenados y transformados en archivo “.Csv”. Fuente Autor.

5.2 SEGMENTACIÓN

Una vez la información se encuentre en el programa Excel se lleva a cabo la segmentación de esta, para esto se dividen los datos en diferentes columnas: Fecha/Hora, temperatura corporal, temperatura ambiente, velocidad, aceleración entre otros ver Ilustración 44. Por último, se importa la información con ayuda del software Matlab, donde se toma una ventana de tiempo de 12 horas para graficar los datos y desarrollar las pruebas estadísticas

A	B	C	D	E	F	G	H	I	J	K
TEMP_CORPORAL(°C)	TEMP_AMBIENTE(°C)	Fecha/Hora	Latitud1	Longitud1	Latitud2	Longitud2	Radio de la t	Distancia (m)	Velocidad(m/s)	Aceleracion
33,83	33,29	13/09 13:20:00	2,472957	-76,532427	2,474002	-76,531392	6378,1	163,65214	0,272753561	-0,000448
34,85	32,81	13/09 13:22:00					6378,1			
33,73	32,93	13/09 13:24:00					6378,1			
32,45	32,73	13/09 13:26:00					6378,1			
35,19	33,31	13/09 13:28:00					6378,1			
33,19	33,17	13/09 13:30:00	2,474002	-76,531392	2,474023	-76,531388	6378,1	2,3796475	0,003966079	2,338E-05
34,51	32,95	13/09 13:32:00					6378,1			
32,13	32,55	13/09 13:34:00					6378,1			
35,05	32,21	13/09 13:36:00					6378,1			
34,69	31,99	13/09 13:38:00					6378,1			
31,81	31,61	13/09 13:40:00	2,474023	-76,531388	2,473978	-76,531302	6378,1	10,796906	0,017994843	0,0037307
30,53	31,45	13/09 13:42:00					6378,1			
33,77	31,27	13/09 13:44:00					6378,1			
31,35	31,07	13/09 13:46:00					6378,1			
34,29	32,65	13/09 13:48:00					6378,1			
35,07	33,41	13/09 13:50:00	2,473978	-76,531302	2,466392	-76,540817	6378,1	1353,8607	2,256434526	-2,647E-05
35,61	34,09	13/09 13:52:00					6378,1			
36,09	34,63	13/09 13:54:00					6378,1			
36,37	35,11	13/09 13:56:00					6378,1			
36,55	35,13	13/09 13:58:00					6378,1			
36,79	35,59	13/09 14:00:00	2,466392	-76,540817	2,473892	-76,531343	6378,1	1344,3333	2,240555461	-0,003729
37,03	35,97	13/09 14:02:00					6378,1			
37,21	36,41	13/09 14:04:00					6378,1			
37,27	36,87	13/09 14:06:00					6378,1			
36,39	36,93	13/09 14:08:00					6378,1			

Ilustración 44. Datos segmentados para el análisis. Fuente Autor.

En la Ilustración 45 se pueden observar las gráficas de la temperatura respecto al tiempo, de los días 13 y 19 de septiembre

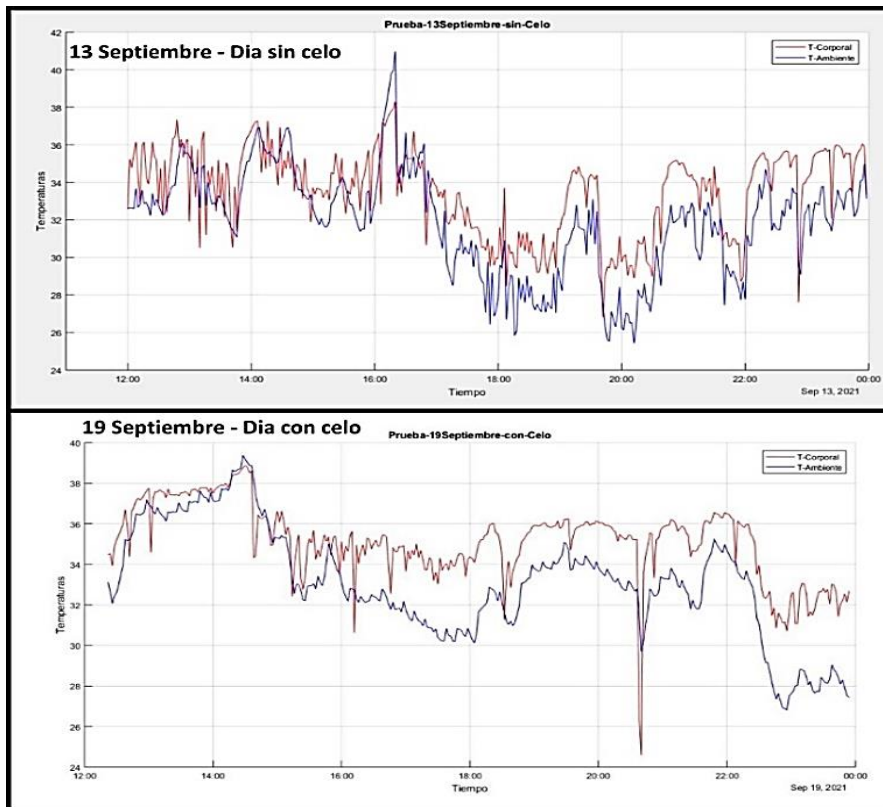


Ilustración 45. Temperaturas obtenidas del 13 y 19 de septiembre. Fuente Autor.

De igual forma se obtuvieron las gráficas de valor absoluto entre la temperatura corporal y ambiental para ambos días. Ver Ilustración 46

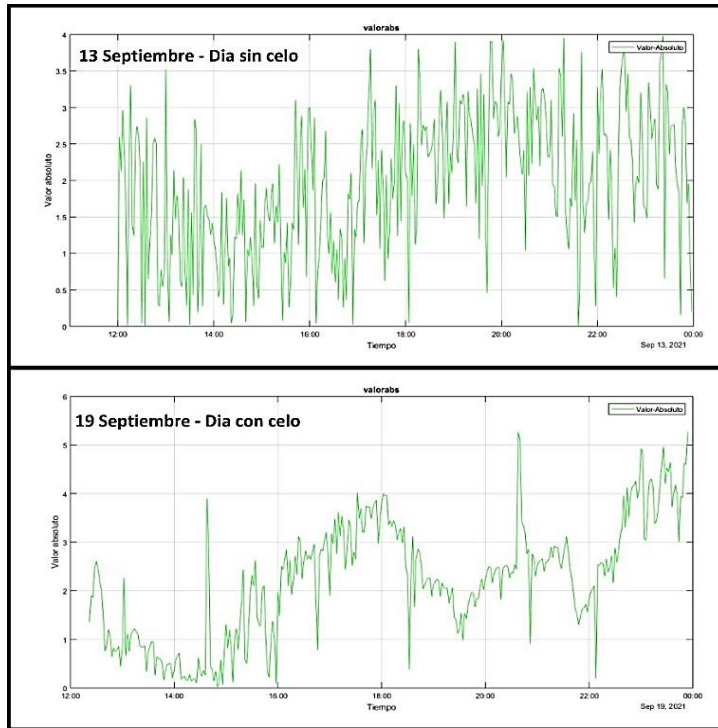


Ilustración 46. Valor absoluto de las temperaturas del 13 y 19 de septiembre. Fuente Autor.

Para la variable aceleración se llevó a cabo el mismo proceso, a continuación, se muestra la gráfica ver Ilustración 47.

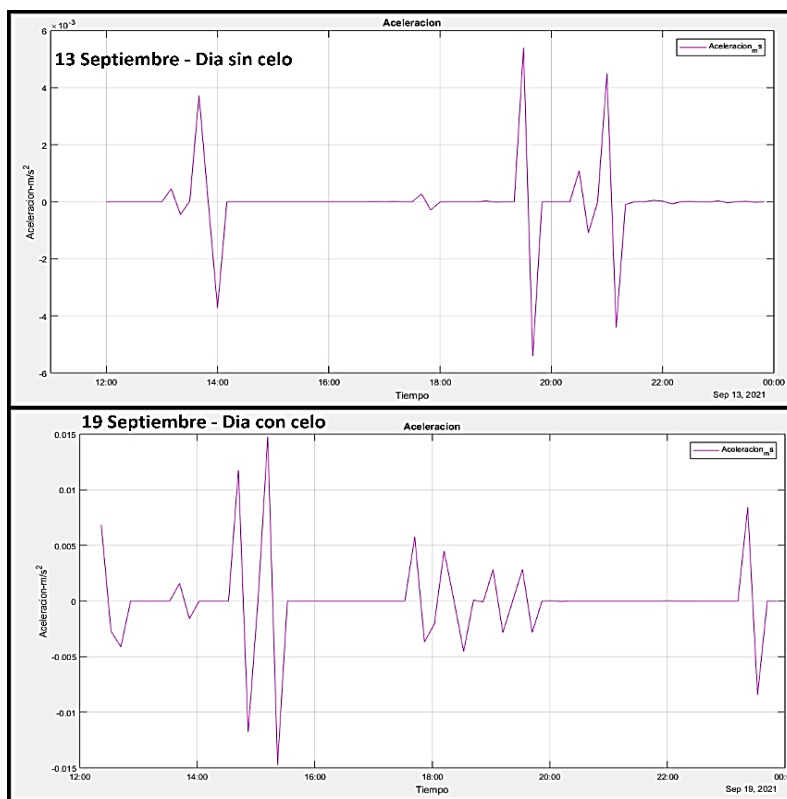


Ilustración 47. Aceleraciones del 13 y 19 de septiembre. Fuente Autor.

5.3 PRUEBAS ESTADÍSTICAS

Para determinar el comportamiento de la variable temperatura, se plantea una serie de pasos recomendados por un experto en el tema de análisis de datos, los cuales se presentan en un mapa conceptual que resume el procedimiento a seguir, ver Ilustración 48

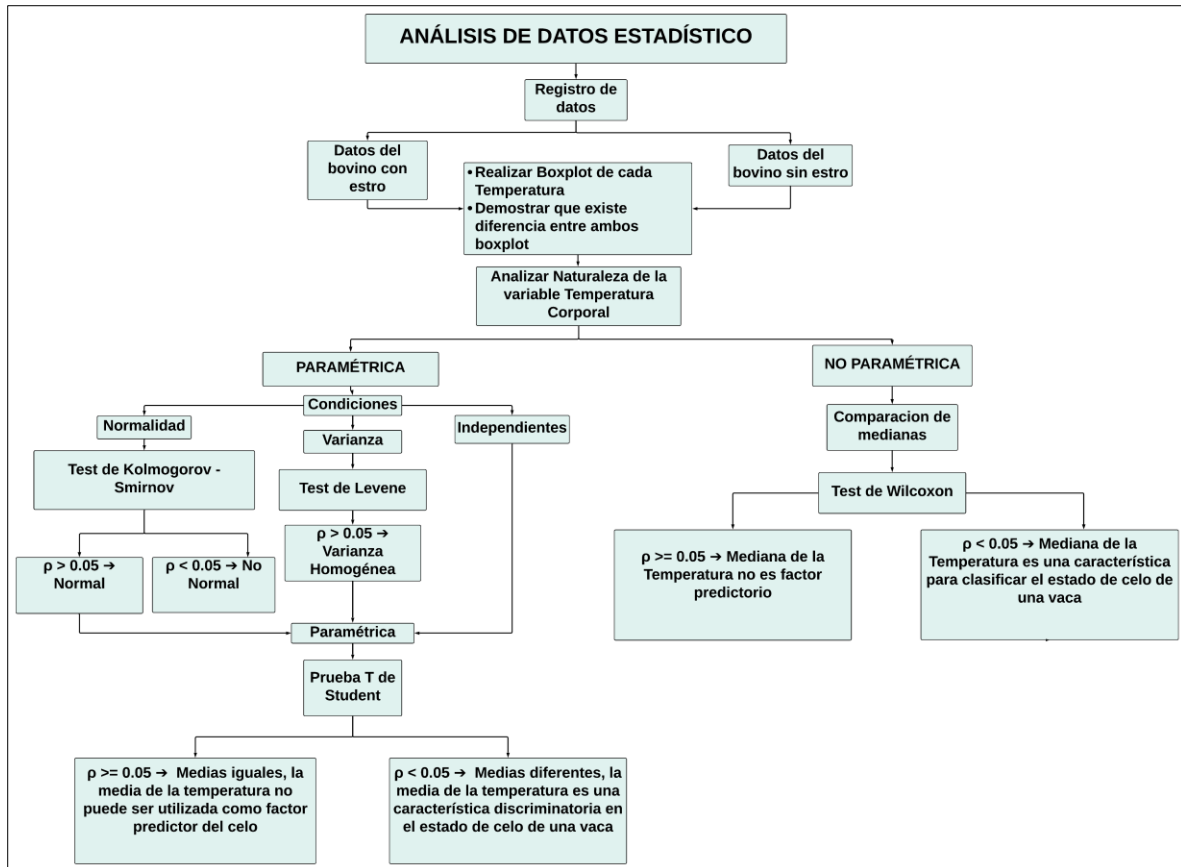


Ilustración 48. Mapa conceptual – Análisis estadístico. Fuente experto en análisis de datos.

A continuación, se describen brevemente las pruebas realizadas para el caso de estudio.

- A partir del registro de datos obtenido en Excel, se definen dos series de datos, ver Ilustración 49.

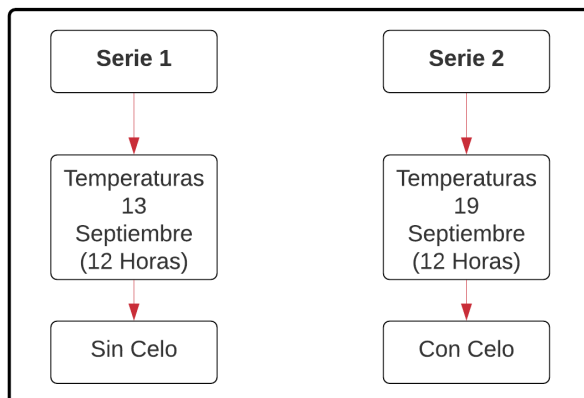


Ilustración 49. Series de temperatura para el análisis. Fuente Autor.

Luego de dividir la información se realiza la comparación de ambas temperaturas, esto con la ayuda del método Boxplot (Diagrama de cajas y bigotes), el cual permite representar de forma gráfica la distribución de puntuaciones dentro de la variable a estudiar [79]. En la Ilustración 50 se muestran los Boxplot de cada serie y se evidencia que visualmente existe diferencia entre los datos de temperatura.

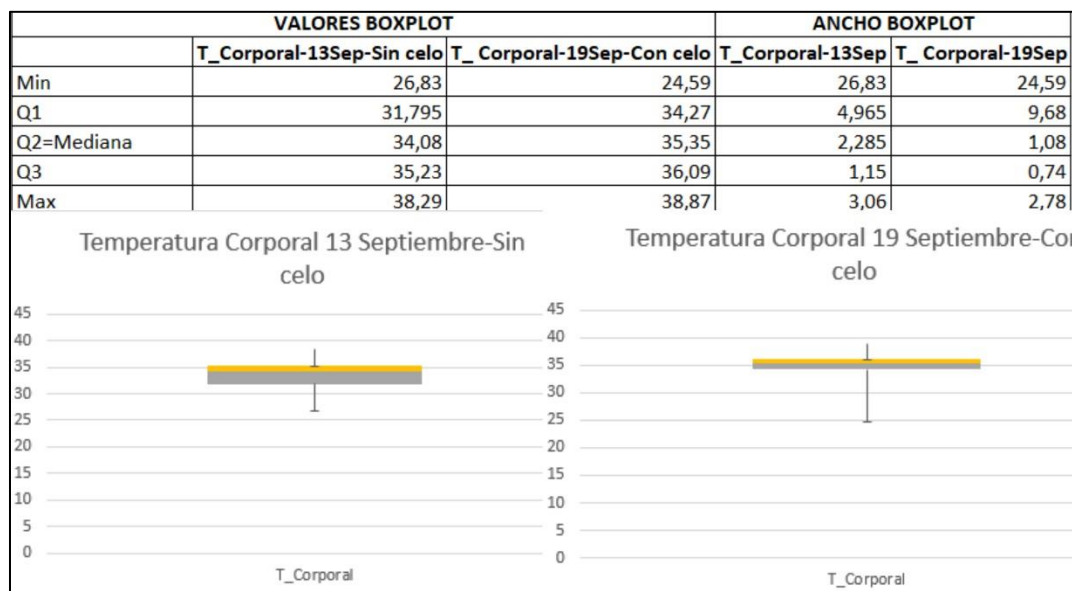


Ilustración 50. Boxplot de las temperaturas del 13 y 19 de septiembre. Fuente Autor.

Entendiendo que existe una diferencia entre los Boxplot de la serie 1 y 2, se continúa con el análisis estadístico en Matlab, por lo que se debe conocer la naturaleza de la variable temperatura es decir si es paramétrica o no paramétrica y así poder comparar las series.

Por lo tanto, para conocer si la variable es paramétrica debe cumplir 3 condiciones, primero que sea normal es decir que se pueda ajustar a una densidad de probabilidad normal, como segundo que sus varianzas sean constantes y por último que sean independientes. Para probar la primera condición se utilizó el test de Kolmogorov que

permite determinar la normalidad de las muestras [80], y saber si los datos son paramétricos o no paramétricos según el p-valor como lo muestra la Ilustración 51.

$$[h, \rho] = kstest(x) \rightarrow \text{Test de Kolmogorov - Smirnov}$$

Condiciones

$$\rho > 0.05 \rightarrow \text{Distribución es normal}$$
$$\rho < 0.05 \rightarrow \text{Distribución es no normal}$$

Ilustración 51. Test de Kolmogorov - Smirnov. Fuente Matlab, [80] .

Cabe mencionar que al cumplirse la normalidad en ambas series y la varianza es homogénea se puede afirmar que las series son paramétricas, por lo tanto, se comparan medias con la ayuda del test T de student.

Para el caso de estudio las series fueron no normales, es decir no paramétricas, por lo que se debe aplicar el test de Wilcoxon que permite comparar las medianas de cada serie [81]. En la Ilustración 52 se muestran las condiciones que debe cumplir el p-valor.

$$[\rho, h] = ranksum(x, y) \rightarrow \text{Test de Wilcoxon}$$

Condiciones

$$\rho \geq 0.05 \rightarrow \text{La mediana de la temperatura no es un factor predictor.}$$
$$\rho < 0.05 \rightarrow \text{La mediana de la temperatura es una característica para clasificar el estado de celo de una vaca.}$$

Ilustración 52. Test de Wilcoxon. Fuente Matlab, [81] .

Este proceso de análisis se realizó también para las variables valor absoluto (temperatura corporal, temperatura ambiente) y aceleración, Los resultados obtenidos en cada una de las pruebas se muestran en la sección 5.4

5.4 RESULTADOS OBTENIDOS ANÁLISIS DE DATOS

En este capítulo se presentan los resultados obtenidos por las pruebas estadísticas del análisis.

- En la Ilustración 45, se puede observar que existe correlación entre las temperaturas (corporal, ambiente), sin embargo, hay un incremento en la temperatura corporal el día 19 de septiembre (día con celo) en comparación con el 13 de septiembre (día sin celo), estos cambios se dan entre las 16:00 y 20:30 (aproximadamente).

- En la Ilustración 46 se visualiza que el día 13 de septiembre presenta variabilidad en la información a comparación con el día 19, en el cual los datos son relativamente más estables, también se puede identificar un incremento del valor absoluto el día 19 alrededor de las 16:00 y 20:30 horas
- La variable aceleración se grafica en la Ilustración 47 donde se comparan dos series de datos (con y sin celo), el día 19 de septiembre se presentan 4 movimientos claros que comienzan a las 15:00, 17:30 y 23:00 horas, por el contrario, el día 13 de septiembre (sin celo), solo se presentan movimientos a las 13:00 y a las 18:00 horas, no fueron representativos, dado que no se alcanza a notar variaciones significativas, por lo tanto no se toma en cuenta.
- Los Boxplot presentados en la Ilustración 50 representan los valores de temperatura correspondientes a los días 13 y 19 de septiembre, al compararlos se evidencia que existe notable diferencia entre los 2, pues visualmente se puede apreciar que las medianas son diferentes, lo que significa que se puede realizar un análisis estadístico más profundo con la variable estudiada.
- Aplicando el test de Kolmogorov, ver Ilustración 51 se obtienen los resultados de la Ilustración 53 donde el p-valor para la temperatura corporal del día 13 de septiembre (sin celo) es de $\rho = 1.8134 \times 10^{-316}$, lo que significa, que la serie de datos 1 no viene de una distribución normal puesto que el p es muy bajo y no cumple con la condición ($p < 0.05$). Para el caso de la serie 2 que representa los valores de temperatura del día 19 de septiembre (sin celo), el resultado de p es $\rho = 3.9833 \times 10^{-305}$, al igual que la serie 1 es no normal. Sabiendo que ambas series no cumplen la condición de normalidad se deduce que la variable es No paramétrica.

NO NORMAL	NO NORMAL
p_Tem13 =	p_Tem19 =
1.8134e-316	3.9833e-305

Ilustración 53. Resultados Test de Kolmogorov de los días 13 y 19 septiembre.

Fuente Autor.

- La Ilustración 54 muestra el valor arrojado por el Test de Wilcoxon, donde se comparan las medianas de ambas series y se obtiene como resultado que $\rho = 3.9907 \times 10^{-22}$, esto indica que cumple con la condición de la prueba ($p < 0.05$) lo que quiere decir que la temperatura corporal del bovino (hembra) es estadísticamente diferente con y sin celo, considerándolo como un factor predictivo.

<p><code>p_Wil_Tem =</code></p> <p><code>3.9907e-22</code></p>
--

Ilustración 54. Resultados Test de Wilcoxon de los días 13 y 19 septiembre. Fuente Autor.

- En la Ilustración 55, se puede observar el resultado obtenido por el parámetro “Left” es de $\rho = 1.9954 \times 10^{-22}$ lo que quiere decir que la temperatura corporal del día 13 de septiembre cuando el bovino (Hembra) no se encontraba en celo, tiene una mediana más baja que la temperatura del 19 de septiembre (día con celo), esto significa que la mediana de la temperatura aumenta cuando el bovino (hembra) está en estro.

LEFT	RIGHT
<p><code>p_Wil_Tem =</code></p> <p><code>1.9954e-22</code></p>	<p><code>p_Wil_Tem =</code></p> <p><code>1</code></p>

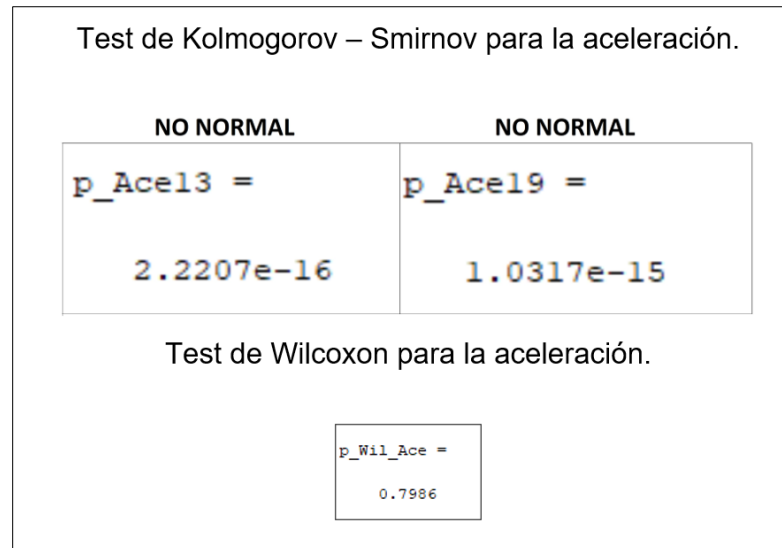
Ilustración 55. Resultados Test de Wilcoxon de los días 13 y 19 septiembre con los parámetros “Left” y “Right”. Fuente Autor.

- Para el caso del valor absoluto se aplicaron los mismos test, en el caso de la prueba de Kolmogorov los resultados fueron no normales, es decir que la variable es no paramétrica. Por otro lado, el test de Wilcoxon arrojó que el valor absoluto es un factor predictor de estado de celo. Ver Ilustración 56.

Test de Kolmogorov – Smirnov para el valor absoluto	
NO NORMAL	NO NORMAL
<p><code>p_V_abs13 =</code></p> <p><code>1.8297e-140</code></p>	<p><code>p_V_abs19 =</code></p> <p><code>3.1487e-126</code></p>
Test de Wilcoxon para el valor absoluto	
LEFT	RIGHT
<p><code>p_Wil_Vabs =</code></p> <p><code>0.0041</code></p>	<p><code>p_Wil_Vabs =</code></p> <p><code>0.9959</code></p>

*Ilustración 56. Resultados del test de Kolmogorov y Wilcoxon para el valor absoluto.
Fuente Autor.*

- En la Ilustración 57, se muestra los resultados arrojados por las pruebas estadísticas a la variable aceleración, indicando que esta no es un factor predictor de celo, pues no cumple con las condiciones de la prueba de Wilcoxon



*Ilustración 57. Resultados del test de Kolmogorov y Wilcoxon para la aceleración.
Fuente Autor.*

5.5 PLANTEAMIENTO DEL ALGORITMO PARA ESTIMAR EL ESTADO DE CELO

En este apartado se presenta el algoritmo modelado mediante un diagrama de flujo, para estimar el celo en ganado bovino (hembra), en primer lugar, se toman muestras de temperaturas corporales durante un tiempo determinado, luego se calcula la mediana x del total de datos. Una vez se tenga este valor se compara con un valor umbral, el cual se obtiene gracias a las pruebas estadísticas. Si la mediana x es mayor que el valor umbral se activa una alarma, que notifique la posible existencia de celo en el animal, en caso contrario si la mediana x es menor, se deduce que el Bovino (hembra) no está en estado de celo.

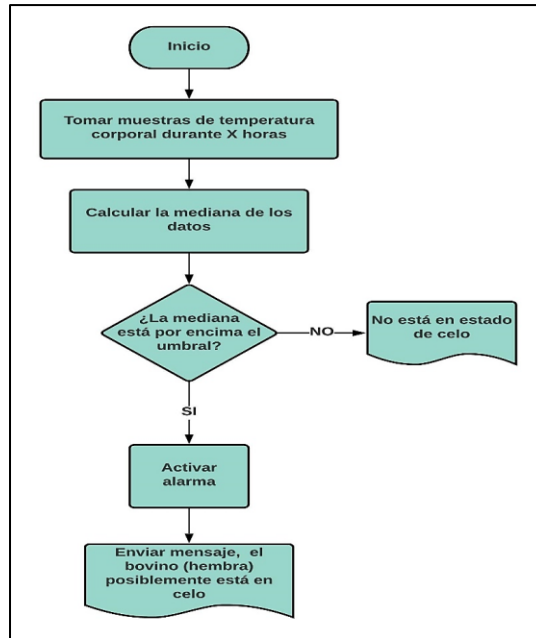


Ilustración 58. Algoritmo estimación de Celo. Fuente Autor

6. CONCLUSIONES

- A partir de los resultados obtenidos con el test de Wilcoxon ver Ilustración 55, se puede afirmar que la temperatura corporal aumenta cuando el bovino (hembra) está en estado de celo, con una probabilidad de error de $1.9954 \times 10^{-20}\%$. Lo cual permite inferir que el sistema IoT así como el monitoreo de las variables de temperatura son un estimador del estado de estro.
- El valor absoluto de la temperatura aumenta cuando la vaca está en celo, con una probabilidad de error de 0.041%.
- La metodología basada en el diseño por requerimientos, permitió seguir de manera ordenada el desarrollo del proyecto, realizando ajustes hasta obtener un prototipo electrónico final, capaz de capturar y enviar los datos a una plataforma, para su posterior análisis.
- La propuesta del algoritmo de estimación de estado de estro, se basa en el aumento de la temperatura corporal puesto que la mediana de esta variable aumenta cuando el bovino (hembra) se encuentra en estado de celo, con un porcentaje de error de $1.9954 \times 10^{-20}\%$. Para generar una alarma desde la plataforma IoT se pueden tomar valores de temperatura durante un lapso de tiempo y se calcula la mediana, si este valor está por encima del umbral se activa una alarma para indicar que puede ocurrir el estro.

7. RECOMENDACIONES Y TRABAJOS FUTUROS

Recomendaciones

- La captura de datos de la variable temperatura se vio afectada por el movimiento del animal, debido a los movimientos bruscos que experimentaba el collar ocasionando que se obtengan datos erróneos. Se sugiere que el prototipo tenga un mejor ajuste en el cuello de animal para esto se puede hacer uso de un arnés o cabezal, que proporciona más fijación.
- El prototipo tuvo filtraciones de agua por la lluvia que se presentaba en la zona de prueba, por esta razón se recomienda impermeabilizar la caja con algún tipo de empaque o otro tipo de aislante.
- Se recomienda verificar el estado de la vaca con la ayuda de un experto en el tema antes de poner el dispositivo, puesto que si presenta problemas de salud o está en estado gestación no entrará en celo.
- Se sugiere verificar la cobertura de la zona, ya que Sigfox es una red que a pesar de tener un gran despliegue, presenta baja señal en algunas zonas rurales.

Trabajos futuros

- Trabajar en una propuesta para mejorar el diseño industrial del dispositivo tomando aspectos como una mejor sujeción, repetibilidad o facilidad de reproducción a escala.
- Integrar un acelerómetro, ya que este ofrecerá más información sobre el comportamiento del animal, permitiendo así generar una propuesta de algoritmo más eficiente a la hora de estimar el estado de celo.
- Implementar una plataforma que permita hacer el análisis de datos, ya que en el presente proyecto se usan diferente software para cada una de las tareas.
- Desarrollar una aplicación móvil que permita visualizar las variables medidas, de igual forma que cuente con la opción de generar alertas acerca del comportamiento del animal, y con esto llevar un monitoreo constante y a distancia.
- Desarrollar un estudio del desempeño del sistema IoT con mayor número de muestras en diferentes bovinos, a modo de contar con un dataset más grande

para analizar distintos comportamientos, y con esto soportar el algoritmo propuesto

8. BIBLIOGRAFÍA

- [1] A. P. Pertuz Martínez and J. E. Elías Caro, "Competitividad en el sector ganadero en Colombia: Enfoque desde la historiografía económica, social y empresarial," *Panor. Económico*, vol. 27, no. 2, pp. 453–480, Mar. 2019, doi: 10.32997/2463-0470-vol.27-num.2-2019-2637.
- [2] Con. Ganadero, "¿Por qué la ganadería es tan importante en Colombia? | CONtexto ganadero | Noticias principales sobre ganadería y agricultura en Colombia," *13 de Junio 2019*, 2019. <https://www.contextoganadero.com/ganaderia-sostenible/por-que-la-ganaderia-es-tan-importante-en-colombia> (accessed Feb. 17, 2020).
- [3] I. C. A. (ICA), "Censo Pecuario año 2019 - ICA," 2019. <https://www.ica.gov.co/areas/pecuaria/servicios/epidemiologia-veterinaria/censos-2016/censo-2018> (accessed Feb. 17, 2020).
- [4] C. de G. del Cauca, "Cifras 2019: Ganadería Departamento del Cauca," p. 5, 2019.
- [5] F. C. de G. (Fedegan), "BASES PARA LA FORMULACIÓN DEL PLAN DE ACCIÓN 2014 – 2018 PARA EL MEJORAMIENTO DE LA GANADERÍA DEL DEPARTAMENTO DE CAUCA," p. 67, 2018, [Online]. Available: <https://estadisticas.fedegan.org.co/DOC/download.jsp?pRealName=7.PlanCaucaFINAL.pdf&ildFiles=653>.
- [6] Gobernación del Cauca, "PLAN DEPARTAMENTAL DE DESARROLLO CAUCA 2016 - 2019," Popayán - Cauca, 2016. Accessed: Sep. 19, 2020. [Online]. Available: <http://www.fundaciondelmacizo.org/wp-content/uploads/2018/03/Plan-de-Desarrollo-2016-2019-Cauca-Territorio-de-Paz.pdf>.
- [7] C. E. Bovinos, S. Dayanna Ortiz Sanabria, and K. Yiney Avila Parra, "FUNDAMENTOS Y MÉTODOS ACTUALES DE DETECCIÓN DE CELO EN BOVINOS," Villavicencio - Meta, May 2020. Accessed: Jul. 23, 2020. [Online]. Available: https://repository.ucc.edu.co/bitstream/20.500.12494/17509/2/2020_fundamentos_metodos_actuales.pdf.
- [8] P. Chen, "Dairy cow health monitoring system based on NB-IoT communication," in *Proceedings - 2019 International Conference on Electronic Engineering and Informatics, EEI 2019*, Nov. 2019, pp. 393–396, doi: 10.1109/EEI48997.2019.00091.
- [9] D. Carrera, "Efectividad de cuatro métodos para la detección de celo en vacuno de carne," p. 76, 2017, [Online]. Available: https://citarea.cita-aragon.es/citarea/bitstream/10532/3955/1/2017_493.pdf.
- [10] J. Hernández Cerón *et al.*, "Fisiología Clínica de la Reproducción de Bovinos LecheRos," Ciudad de México, Nov. 2016. Accessed: Feb. 22, 2021. [Online]. Available: https://fmvz.unam.mx/fmvz/publicaciones/archivos/Fisiologia_Clinica.pdf?fbclid=IwAR3n1j-6OsjeRJvty4huWvCjRSkkDVTb-jzPsrtPzkS0vxySWk8G4T4EMjQ.
- [11] T. Nacional, "MANEJO PRODUCTIVO Y REPRODUCTIVO EN BOVINOS, OVINOS, CAPRINOS Y EQUINOS," 2018. Accessed: Jul. 23, 2020. [Online]. Available: https://www.tecnacional.edu.ni/media/Manual_Bovino_y_Caprino_opt.pdf.
- [12] S. Montero Hernández, "Análisis del estado actual de las tecnologías de sensorización inalámbrica aplicadas a la gestión eficiente de rebaños de vacuno

- de carne en extensivo,” p. 119, 2019, [Online]. Available: <http://uvadoc.uva.es/bitstream/handle/10324/38599/TFM-L436.pdf?sequence=1&isAllowed=y>.
- [13] R. Adeva, “Qué es el GPS y otros sistemas de navegación por satélite,” *Roberto Adeva Publicado el 10 de abril, 2020 • 20:00 adslzone*, Apr. 10, 2020. <https://www.adslzone.net/reportajes/tecnologia/gps-que-es-redes/> (accessed Jul. 23, 2020).
- [14] “Sistema GPS, su historia y evolución en el tiempo - GPS satelital.org,” Jan. 08, 2019. <https://gpsatelital.org/gps-historia-y-evolucion/> (accessed Jul. 23, 2020).
- [15] Fondear, “NMEA; para poder compartir toda la electrónica que tenemos en el barco,” 2017. http://www.fondear.org/infonautic/Equipo_y_Usos/Electronica_Instrumentacion/Protocolo-Datos/NMEA.asp (accessed Jul. 23, 2020).
- [16] J. Jordán, M. Jiménez, and M. Montesinos, “Nueva librería de acceso a dispositivos de localización libLocation,” Valencia - España. Accessed: Jul. 23, 2020. [Online]. Available: http://downloads.gvsig.org/download/events/jornadas-lac/1as-jornadas-lac/articulos/Articulo-Nueva_libreria_acceso_dispositivos_localizacion_libLocation.pdf.
- [17] A. Fernández Sierra, “Utilidades de Matlab para la gestión de los datos crudos obtenidos por un receptor GPS con tecnología SiRF ,” 2013.
- [18] T. Nanayakkara, F. Sahin, and M. Jamshidi, “Intelligent Control Systems with an Introduction to System of Systems ... - Thrishantha Nanayakkara, Ferat Sahin, Mo Jamshidi - Google Libros,” 2010. <https://books.google.com.co/books?id=gH50DwAAQBAJ&pg=SA10-PA86&lpg=SA10-PA86&dq=protocol+tsip&source=bl&ots=8TIs9AkWqC&sig=ACfU3U3tBGWNSPXCLRdjJ0KyguDKZR-wrA&hl=es-419&sa=X&ved=2ahUKEwitw5P2-NfpAhXCJt8KHTzfBt4Q6AEwBHoECAkQAQ#v=onepage&q=protocol+tsip&f=false> (accessed Jul. 23, 2020).
- [19] N. Molina, “Diseño de un prototipo para la identificación de drones mediante el sistema AIS,” 2017.
- [20] A. E. Riba, J. Damián Tejada, N. Acosta, and J. M. Toloza, “Procesamiento de sentencias NMEA-0183 para el análisis de la geometría satelital utilizando receptores GPS de bajo costo.” Accessed: Jul. 23, 2020. [Online]. Available: http://sedici.unlp.edu.ar/bitstream/handle/10915/56419/Documento_completo.pdf-PDFA.pdf?sequence=1&isAllowed=y.
- [21] Jorge Alambiaga Pascual, “ADQUISICIÓN Y PROCESADO DE INFORMACIÓN DE POSICIONAMIENTO GPS MEDIANTE DISPOSITIVO INALÁMBRICO BASADO EN ARDUINO,” Sep. 2017. Accessed: Jul. 23, 2020. [Online]. Available: https://riunet.upv.es/bitstream/handle/10251/88976/ALAMBIAGA_Adquisición_y_procesado_de_información_de_posicionamiento_GPS_mediante_dispositivo_in....pdf?sequence=1.
- [22] Sigma Electrónica, “Como interpretar una trama GPS - Sigma Electrónica,” *Sigma Electrónica*, Oct. 05, 2016. <https://www.sigmaelectronica.net/trama-gps/> (accessed Jul. 23, 2020).
- [23] Sigfox España, “Sigfox .” <https://www.sigfox.es/> (accessed Jul. 23, 2020).
- [24] LoRaWAN, “LoRaWAN.” <https://lorawan.es/> (accessed Jul. 23, 2020).
- [25] L. Dopazo González, “Test y despliegue de tecnología de comunicaciones LoRa

- para aplicaciones de Internet of Things,” Madrid, 2019. Accessed: Jul. 02, 2021. [Online]. Available: http://oa.upm.es/54465/1/TFG_LAURA_DOPAZO_GONZALEZ.pdf.
- [26] the things network, “the things network.” <https://www.thethingsnetwork.org/>.
- [27] Zigbee, “Zigbee - Zigbee Alianza,” Nov. 07, 2019. <https://zigbeealliance.org/es/solución/Zigbee/> (accessed Jul. 23, 2020).
- [28] J. Cruz, “ZigBee para IoT,” Aug. 26, 2018. <https://medium.com/análisis-de-la-tecnología-zigbee-para-su-uso-en-el/zigbee-para-iot-12666b636821> (accessed Jul. 23, 2020).
- [29] wi-fi, “Wi-Fi.” <https://www.wi-fi.org/discover-wi-fi> (accessed Jul. 23, 2020).
- [30] “What Is a Wireless Network? .” <https://www.conceptdraw.com/How-To-Guide/what-is-a-wireless-network> (accessed Jul. 23, 2020).
- [31] efor, “TECNOLOGÍAS DE COMUNICACIÓN PARA IOT.” Accessed: Jul. 23, 2020. [Online]. Available: <https://www.efor.es/sites/default/files/tecnologias-de-comunicacion-para-iot.pdf>.
- [32] Digitanimal, “Características del dispositivo Digitanimal,” *Digitanimal*. <https://dqzrr9k4bjpk.cloudfront.net/images/11723261/1351311732.jpg> (accessed Jul. 23, 2020).
- [33] J. A. Ruiz Somoyar, “MONITOREO EN LA GANADERIA BOVINA A TRAVÉS DE REDES GSM/GPRS,” Medellín, 2016. Accessed: Apr. 09, 2021. [Online]. Available: [https://repository.usta.edu.co/bitstream/handle/11634/1126/Monitoreo en la ganaderia bovina a traves de redes gsm gprs.pdf?sequence=1](https://repository.usta.edu.co/bitstream/handle/11634/1126/Monitoreo%20en%20la%20ganaderia%20bovina%20a%20traves%20de%20redes%20gsm%20gprs.pdf?sequence=1).
- [34] Silicon Labs, “Datasheet Si7050/1/3/4/5-A20 I 2 C TEMPERATURE SENSORS Features Applications Description Pin Assignments Top View,” 2018. Accessed: Apr. 10, 2021. [Online]. Available: <https://www.silabs.com/documents/public/data-sheets/Si7050-1-3-4-5-A20.pdf>.
- [35] TE Connectivity company, “Datasheet TSYS01 Digital Temperature Sensor,” Jun. 2018. Accessed: Apr. 10, 2021. [Online]. Available: https://www.te.com/commerce/DocumentDelivery/DDEController?Action=show_doc&DocId=Data+Sheet%7FTSYS01%7FA%7Fpdf%7FEnglish%7FENG_DS_TSYS01_A.pdf%7FG-NICO-018.
- [36] Melexis, *MLX90614 Datasheet Single and Dual Zone Infrared Thermometer in TO-39*. 2018.
- [37] Maxim integrated, “Datasheet DS18B20+,” 2019. Accessed: Apr. 10, 2021. [Online]. Available: <https://co.mouser.com/datasheet/2/256/DS18B20-370043.pdf>.
- [38] Quectel, “Datasheet L80-R Hardware Design GPS Module Series,” 2016. Accessed: Apr. 09, 2021. [Online]. Available: http://www.dragino.com/downloads/downloads/datasheet/other_vendors/L80-R/Quectel_L80-R_Hardware_Design_V1.2.pdf.
- [39] Quectel, “Datasheet L86 Hardware Design GNSS Module Series,” 2014. Accessed: Apr. 10, 2021. [Online]. Available: <https://docs.rs-online.com/1fe5/0900766b8147dbfb.pdf>.
- [40] Ublox, “Datasheet CAM-M8 u-blox M8 .” Accessed: Apr. 10, 2021. [Online]. Available: https://www.u-blox.com/sites/default/files/CAM-M8-FW3_DataSheet_%28UBX-15031574%29.pdf.
- [41] Quectel, “Datasheet Quectel L96 Extremely Compact GNSS Module with Chip Antenna,” 2017. Accessed: Apr. 10, 2021. [Online]. Available: https://co.mouser.com/datasheet/2/1052/Quectel_L96_GNSS_Specification_V

- 1.1-1829976.pdf.
- [42] Microchip, "Datasheet SAM D21," 2020. Accessed: Feb. 04, 2021. [Online]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/SAM_D21_DA1_Family_DataSheet_DS40001882F.pdf.
- [43] Atmel, "Datasheet ATmega328p." Accessed: Feb. 04, 2021. [Online]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf.
- [44] Microchip, *DataSheet PIC24F08KA101* . 2008.
- [45] Microchip, "Datasheet PIC24f08KM204 ," 2020. Accessed: Apr. 10, 2021. [Online]. Available: <https://ww1.microchip.com/downloads/en/DeviceDoc/PIC24FV16KM204-Family-Data-Sheet-DS30003030C.pdf>.
- [46] T. Dicola, "Overview | MicroPython for SAMD21 | Adafruit Learning System." <https://learn.adafruit.com/micropython-for-samd21> (accessed Feb. 22, 2021).
- [47] P. A. Montalvo Aguilar, "Desarrollo de un módulo lector braille electrónico para personas con discapacidad visual orientado a portabilidad y ergonomía," Quito, 2021. Accessed: Jul. 02, 2021. [Online]. Available: <http://dspace.ups.edu.ec/handle/123456789/19987>.
- [48] Art Pini and Digikey, "Por qué y cómo usar la interfaz periférica serial para simplificar las conexiones entre distintos dispositivos," 2019-02-14, Feb. 14, 2019. <https://www.digikey.com/es/articles/why-how-to-use-serial-peripheral-interface-simplify-connections-between-multiple-devices> (accessed Jul. 02, 2021).
- [49] D. A. Alvarado Fernández and J. E. Deza Concori, "Diseño e Implementación de una Tarjeta Electrónica de Expansión Modular Basado en PSoC para el Desarrollo de Aplicaciones de Sistemas Embebidos," UNIVERSIDAD PRIVADA DE TACNA, 2019.
- [50] Components101, "18650 Lithium Ion Battery," 08-06-2018, Jun. 08, 2018. <https://components101.com/batteries/18650-lithium-cell> (accessed Jul. 02, 2021).
- [51] L. SEONG JI INDUSTRIAL CO., "Wisol Datasheet WSSFM11R2DAT." Accessed: Apr. 06, 2021. [Online]. Available: https://storage.googleapis.com/public-assets-xd-sigfox-production-338901379285/j5i7saxm_ds_wssfm11r2dat_rev01_170709.pdf.
- [52] L. F. Noguera Lomas, "SISTEMA DE ALERTA TEMPRANA DE INUNDACIONES MEDIANTE UNA RED DE SENSORES INALAMBRICOS LPWAN DIRIGIDO A LA PARROQUIA DE ANDRADE MARÍN PERTENECIENTE AL CANTON ANTONIO ANTE.," UNIVERSIDAD TÉCNICA DEL NORTE, 2020.
- [53] Didácticas Electrónicas, "Batería Litio-Polímero 3.7V 850mAh." <https://www.didacticaselectronicas.com/index.php/baterias-cargadores/recargables-1/tipo-lipo-1/3-7v/bateria-litio-polimero-3-7v-850mah-ion-pol-357090-3-7-2500ma-baterias-pilas-de-litio-polimero-de-3-7v-1s-recargables-detail> (accessed Jul. 02, 2021).
- [54] R. Components, "11 redes inalámbricas fundamentales para Internet de las Cosas," 2016. Accessed: Feb. 26, 2020. [Online]. Available: <https://www.redeweb.com/ficheros/articulos/p62a65.pdf>.
- [55] O. O. Kazeem, O. O. Akintade, and L. O. Kehinde, "Comparative Study of Communication Interfaces for Sensors and Actuators in the Cloud of Internet of

- Things,” *Int. J. Internet Things*, vol. 6, no. 1, pp. 9–13, 2017, doi: 10.5923/j.ijit.20170601.02.
- [56] C. F. Farfán, A. Corena, and B. Colombia, “INGENIERÍA BÁSICA, CONCEPTUAL, DETALLADA Y PROCURA DE EQUIPOS PARA LA IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO DE SIGNOS VITALES DE BOVINOS EN FINCAS GANADERAS.,” Bogotá, 2016. Accessed: Jun. 22, 2021. [Online]. Available: <https://repository.usta.edu.co/bitstream/handle/11634/2372/Farfancamilo2016.pdf?sequence=1>.
- [57] LoRa Alliance Technical Committee Regional Parameters Workgroup, “RP002-1.0.3 LoRaWAN® Regional 41 Parameters,” May 2021. Accessed: Jun. 22, 2021. [Online]. Available: <https://lora-alliance.org/wp-content/uploads/2021/05/RP-2-1.0.3.pdf>.
- [58] Sigfox, “Sigfox Cobertura.” <https://backend.sigfox.com/welcome/coverage> (accessed Jul. 02, 2021).
- [59] Y. C. Flores Banchón and J. C. Maza Pacheco, “DISEÑO DE UNA RED INALÁMBRICA DE MONITOREO DE CO2 BASADO EN LA TECNOLOGIA SIGFOX,” Guayaquil, 2018. Accessed: Apr. 09, 2021. [Online]. Available: <https://www.dspace.espol.edu.ec/retrieve/132286/D-CD106677.pdf>.
- [60] Sigfox, “Radio Configurations | Sigfox build,” 2018. <https://build.sigfox.com/sigfox-radio-configurations-rc> (accessed Jul. 02, 2021).
- [61] H. Micron, “Datasheet iMCP HT32SX V2.2 – SiP Sigfox,” [Online]. Available: <https://partners.sigfox.com/products/imcp-ht32sx>.
- [62] InnoComm, “Datasheet SN10-11 Module for your SIGFOX application.”
- [63] InnoComm, “Datasheet SN10-12 Module for your SIGFOX application.” Accessed: Apr. 06, 2021. [Online]. Available: https://storage.googleapis.com/public-assets-xd-sigfox-production-338901379285/jtldwohv_brochure_sn10-1x_v2.4_en_sn10-12.pdf.
- [64] EAGLE, “EAGLE | PCB Design And Electrical Schematic Software | Autodesk.” <https://www.autodesk.com/products/eagle/overview?term=1-YEAR> (accessed Jul. 02, 2021).
- [65] Arduino, “Arduino - RTCZero,” Dec. 24, 2019. <https://www.arduino.cc/en/Reference/RTC> (accessed Jul. 02, 2021).
- [66] Siemens, “Solid Edge | Siemens | Diseño en 3D, simulación, fabricación.” <https://solidedge.siemens.com/es/> (accessed Jul. 02, 2021).
- [67] P. V. Hasler Mardonez, “Sistema de gestión de ventas en terreno para dispositivos móviles utilizando plataforma como servicio en nube, para la Distribuidora ‘El Gato,’” UNIVERSIDAD DEL BÍO – BÍO, Chillan, Chile, 2016.
- [68] Arduino, “Arduino - Home.” <https://www.arduino.cc/> (accessed Feb. 22, 2021).
- [69] “IoT platform | Internet of Things | Ubidots.” <https://ubidots.com/> (accessed Feb. 22, 2021).
- [70] “IoT Analytics - ThingSpeak Internet of Things.” <https://thingspeak.com/> (accessed Feb. 22, 2021).
- [71] “IoT Hub | Microsoft Azure.” <https://azure.microsoft.com/es-es/services/iot-hub/> (accessed Feb. 22, 2021).
- [72] “AWS | Cloud Computing - Servicios de informática en la nube.” <https://aws.amazon.com/es/> (accessed Feb. 22, 2021).
- [73] Ayusharma0698, “IEEE Standard 754 Floating Point Numbers - GeeksforGeeks,” Mar. 16, 2020. <https://www.geeksforgeeks.org/ieee-standard-754-floating-point-numbers/> (accessed Jul. 02, 2021).

- [74] B. Oviedo Bayas and E. Samaniego Mena, *Fundamentos de informática*, 1st ed. Guayaquil-Ecuador, 2018.
- [75] Wisol, "AT Command Modulo Wisol," pp. 1–6.
- [76] Ubidots, "UbiFunctions User Guide | Ubidots Help Center." <https://help.ubidots.com/en/articles/2132086-analytics-ubifunctions-user-guide> (accessed Apr. 10, 2021).
- [77] "Imagen mlx90614." <https://dynamoelectronics.com/wp-content/uploads/2020/07/mlx-temperatura-compressor.png> (accessed Apr. 10, 2021).
- [78] Openelectronics, "GPS L80-R module with antenna - Open Electronics - Open Electronics." <https://www.open-electronics.org/gps-l80-r-module-with-antenna/> (accessed Apr. 09, 2021).
- [79] B. Rodrigues De Oliveira, "ANÁLISE DE DADOS UTILIZANDO GRÁFICOS BOXPLOT," 2016, Accessed: Oct. 23, 2021. [Online]. Available: https://d1wqtxts1xzle7.cloudfront.net/57148727/TCC-with-cover-page-v2.pdf?Expires=1635013404&Signature=Etc0ehUj4BFwN9oAsRwo1cgWAWm-9de~9UPeLXjgBHsxJ6e1v2qmeCVqdbgkClo9~8tSUmqw3V33dvMLvA4aa-fQrSelp8WQEK1rIRVkkVkos98-B5xyE~TVua7VAjkia33hbg36RaWOt5sgM~~0PX3vQla2IAihzMsPUQlv4-QqJqShT~irRHWFgm5N6b8duSu-UXDyocJ7LVRO3y0~iFLav2CWAC2KOYzpyBIDV0Fb6W0pE~JSV83yLC3TtmP1e-b3dBEbKfIJWPK8sGjeOdgUo2vMLwtqQAbmLHLivOnWLqNw92IC68mk9jRubY2BJHVFwazEKuDWATtG~pse-g__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA.
- [80] S. E. Arévalo Cabrera and C. R. Santisteban Morales, "Modelo de Pronóstico de la Cantidad de Producción de Electricidad con la Metodología de Box Jenkins de la Región Lambayeque en el periodo 2005–2017," UNIVERSIDAD NACIONAL PEDRO RUIZ GALLO, Lambayeque, 2020.
- [81] Minitab, "Interpretar los resultados clave para Wilcoxon de 1 muestra," 2019, Accessed: Oct. 23, 2021. [Online]. Available: <https://support.minitab.com/es-mx/minitab/18/help-and-how-to/statistics/nonparametrics/how-to/1-sample-wilcoxon/interpret-the-results/key-results/>.

ANEXOS

Anexo A. Configuración del backend de Sigfox para enviar datos a Ubidots.

Para enviar datos desde el backend de sigfox a ubidots, es necesario configurar ambas plataformas previamente, por lo que inicialmente se explicará cómo visualizar la carga útil, lo cual ayudará a que los datos se muestren en su formato de origen es decir en hexadecimal, y luego se enumeran los pasos para la creación y configuración del callback.

Visualización carga útil.

1. Dar clic en device type

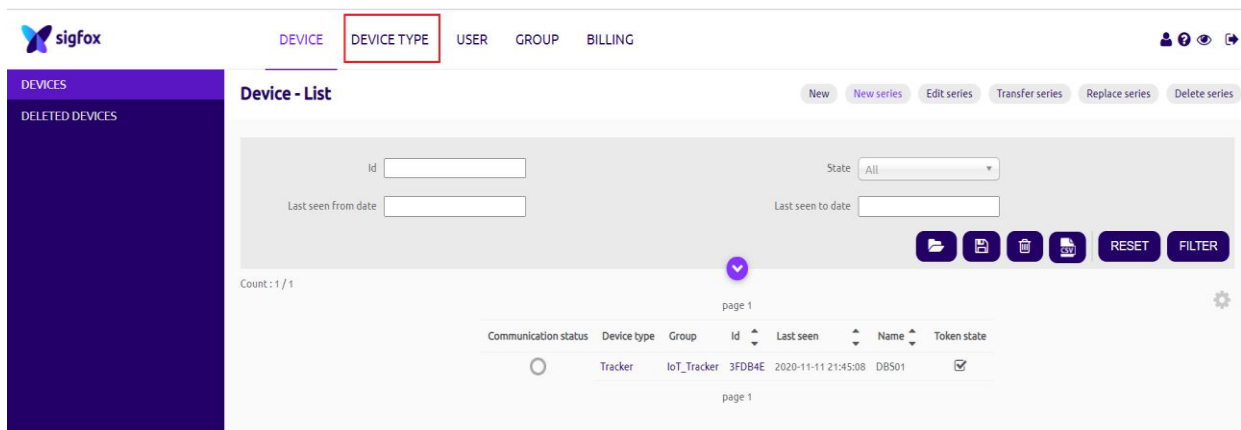


Ilustración 59. Configuración Backend Sigfox - Device Type. Fuente Autor.

2. Seleccionar el Device nombre del dispositivo como aparece a continuación

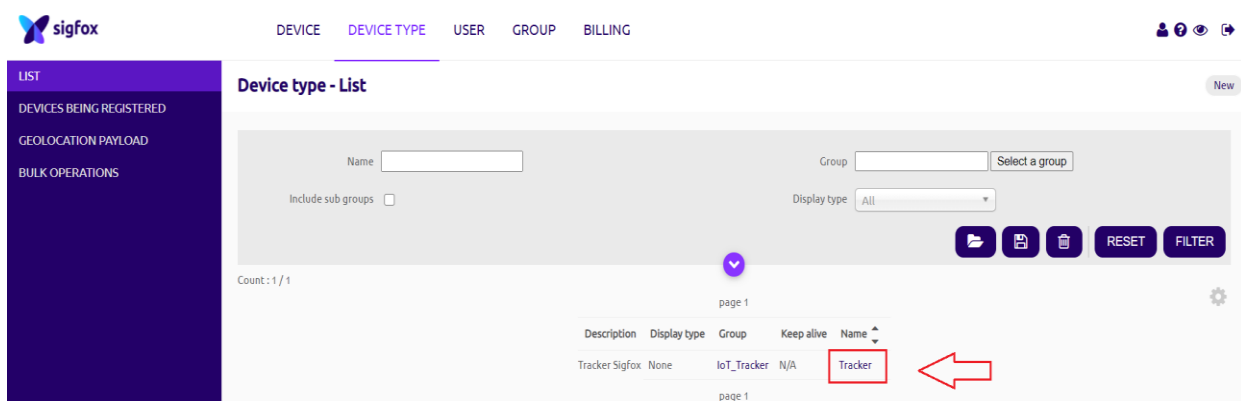


Ilustración 60. Configuración Backend Sigfox - Device Type Name. Fuente Autor.

3. Después de elegir el nombre del dispositivo, se debe dar clic en editar

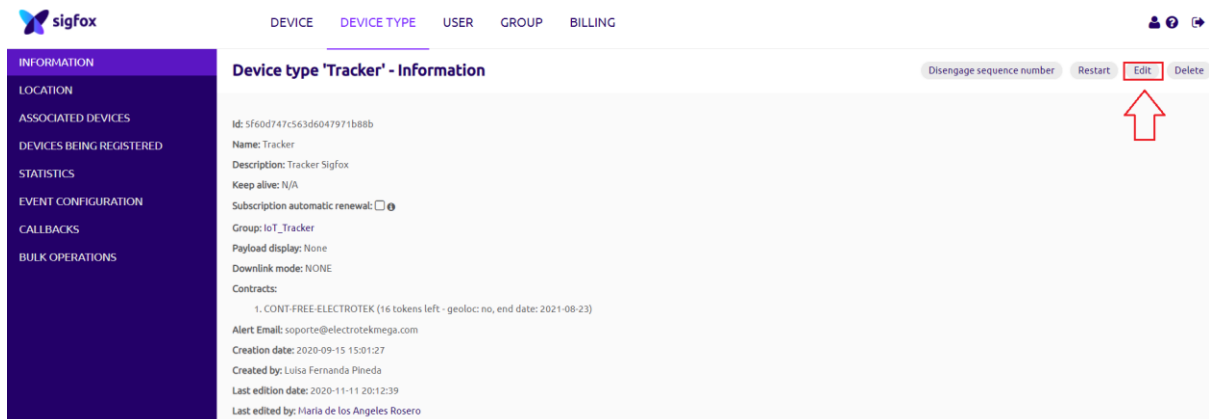


Ilustración 61. Configuración Backend Sigfox - Device Type Edit. Fuente Autor.

4. Se debe establecer la forma en que los datos serán mostrados por el backend de Sigfox, dando clic en la opción Regular (raw payload). Con este parámetro la carga útil no se decodifica y se visualiza con el formato de origen, que para este caso es hexadecimal.

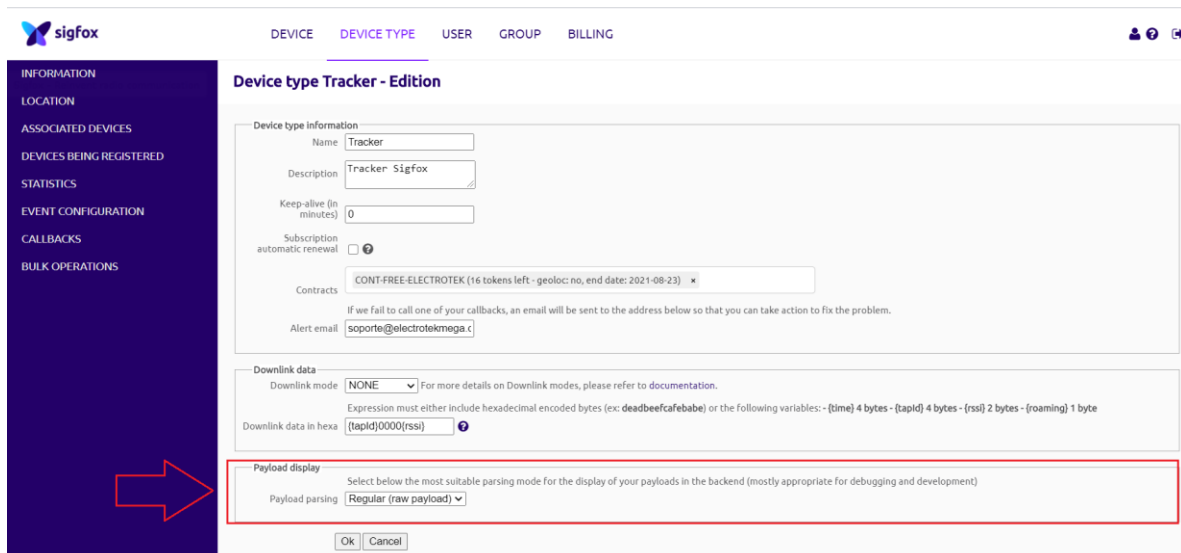


Ilustración 62. Configuración Backend Sigfox - Device Type Payload Display. Fuente Autor.

Para poder establecer una comunicación entre el backend de Sigfox y un servidor externo se debe hacer uso de un callback.

Creación callback

1. Dar clic en device type

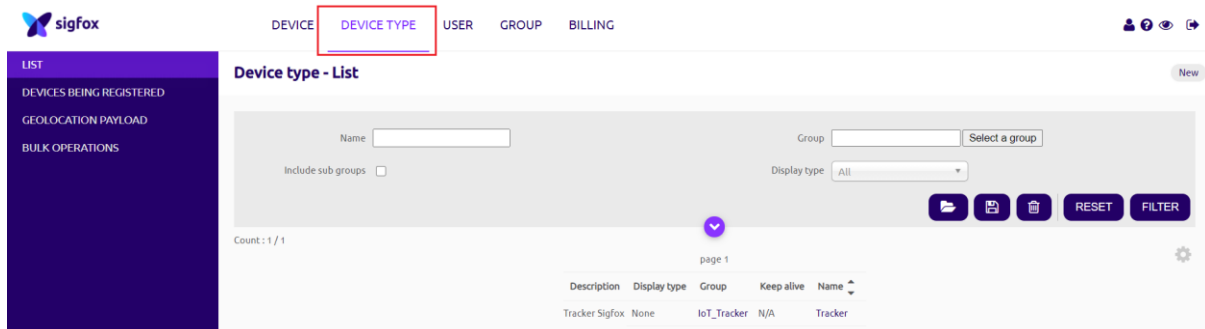


Ilustración 63. Configuración Backend Sigfox - Device Type CallBack . Fuente Autor.

2. Seleccionar el nombre del dispositivo como aparece a continuación

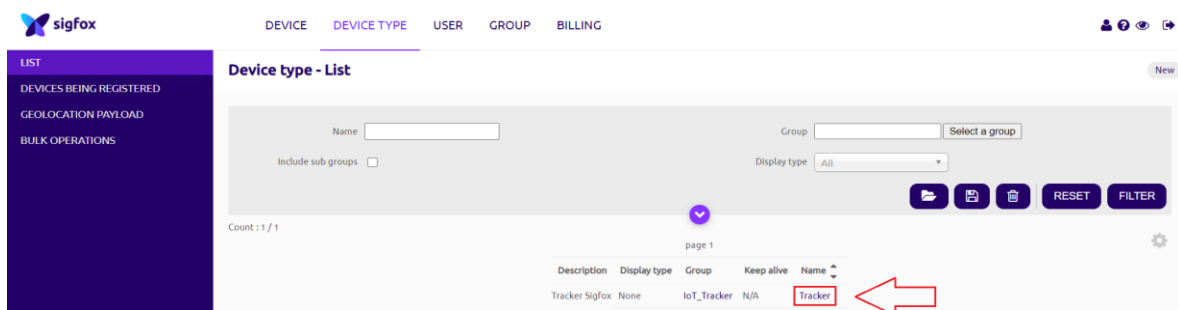


Ilustración 64. Configuración Backend Sigfox - Device Type Callback Name. Fuente Autor.

3. Se elige la opción callback, ubicada en la parte izquierda de la plataforma.

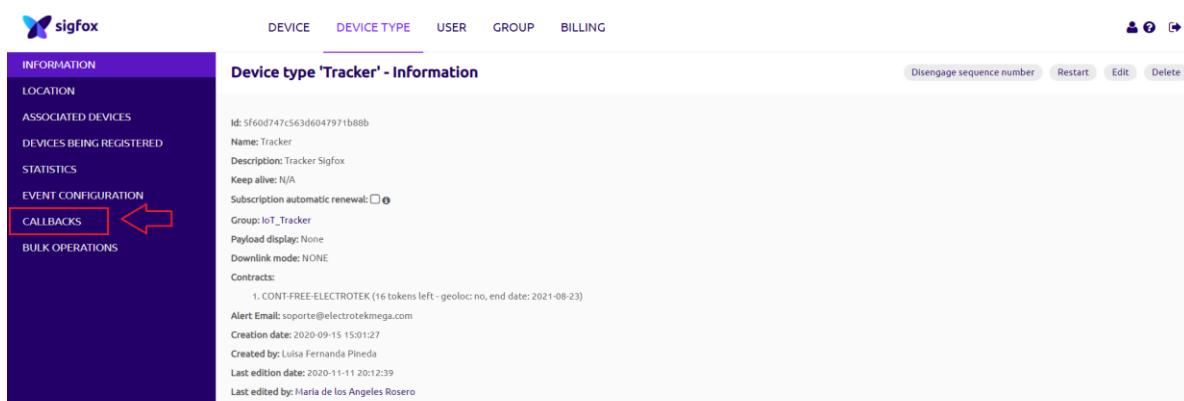


Ilustración 65. Configuración Backend Sigfox - Callback Seleccin. Fuente Autor.

4. Elegir la opción nuevo

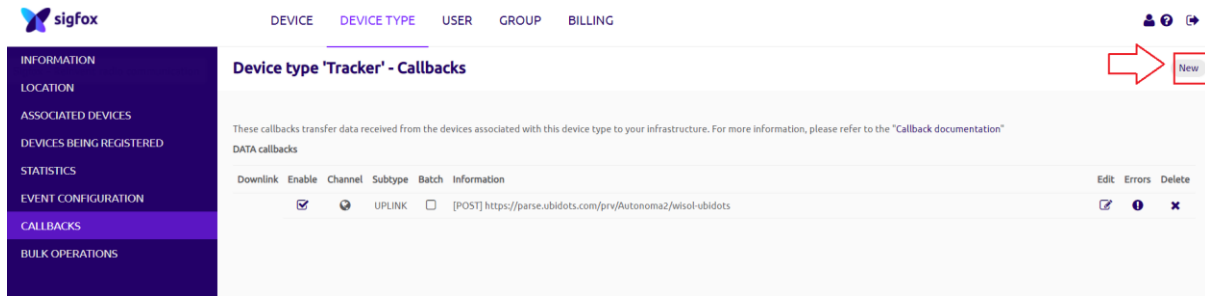


Ilustración 66. Configuración Backend Sigfox - Callback Nuevo. Fuente Autor.

5. Se debe seleccionar la opción custom callback.

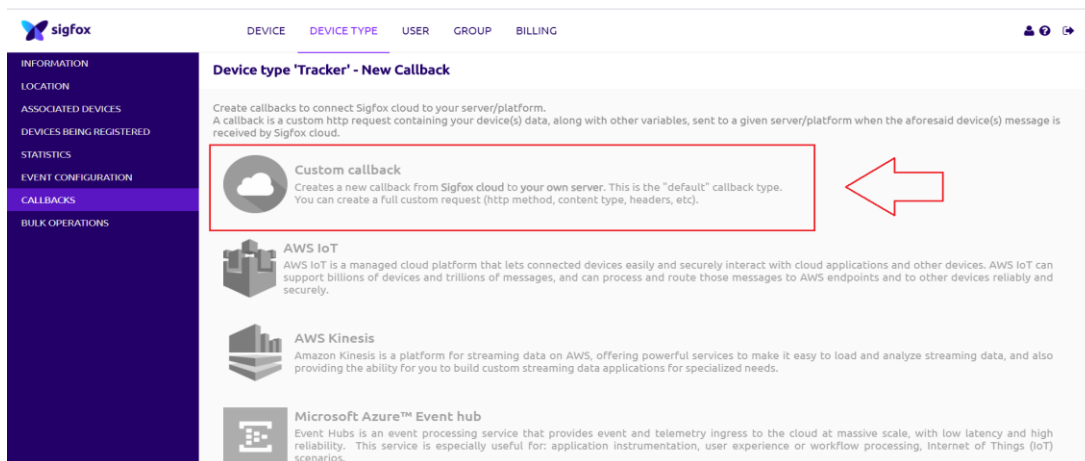


Ilustración 67. Configuración Backend Sigfox - Custom Callback. Fuente Autor.

6. Completar los datos de acuerdo a la siguiente imagen.

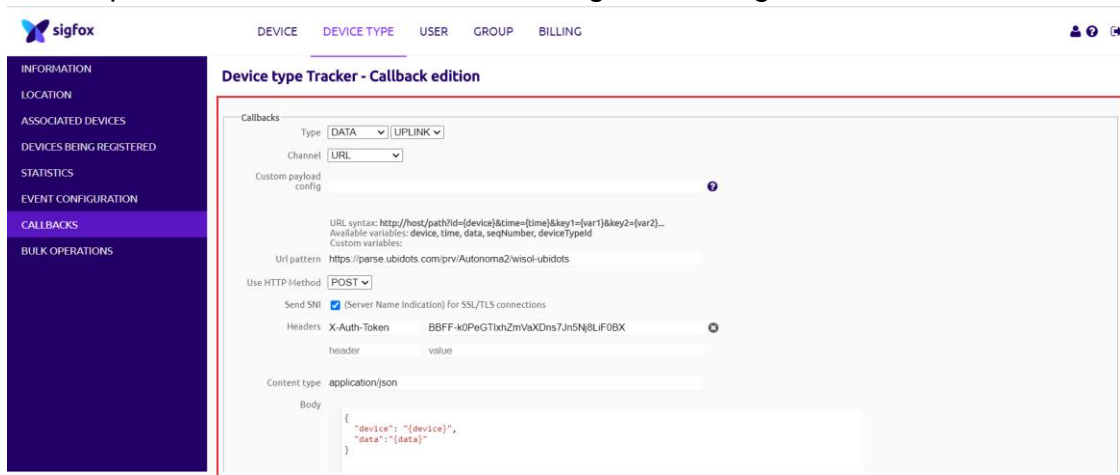
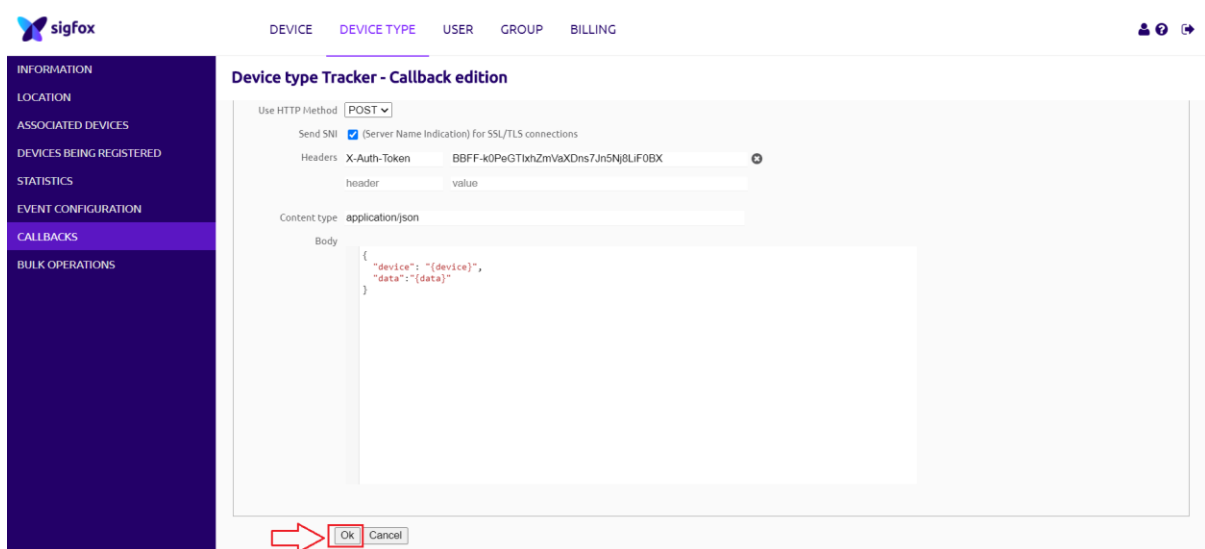


Ilustración 68. Configuración Backend Sigfox - Configuración Callback. Fuente Autor.

7. Clic en la opción ok, ubicada en la parte final de la página.



The screenshot shows the Sigfox Device type Tracker - Callback edition configuration page. The page has a dark blue sidebar on the left with navigation options: INFORMATION, LOCATION, ASSOCIATED DEVICES, DEVICES BEING REGISTERED, STATISTICS, EVENT CONFIGURATION, CALLBACKS (highlighted), and BULK OPERATIONS. The main content area is titled "Device type Tracker - Callback edition" and contains a form for configuring a callback endpoint. The form includes a "Use HTTP Method" dropdown set to "POST", a "Send SNI" checkbox checked, and a "Headers" section with "X-Auth-Token" set to "BBFF-k0PeGTlxhZmVaXDns7Jn5Nj8LlF0EX". The "Content type" is set to "application/json". The "Body" field contains a JSON object:

```
{  "device": "{device}",  "data": "{data}"}
```

. At the bottom of the form, there are "Ok" and "Cancel" buttons, with a red arrow pointing to the "Ok" button.

Ilustración 69. Configuración Backend Sigfox - Callback Fin. Fuente Autor.

Anexo B. Configuración de la plataforma Ubidots.

En este apartado se explican los pasos a seguir, para configurar desde cero la plataforma ubidots, con el fin de poder visualizar los datos provenientes desde el backend de Sigfox.

Creación cuenta ubidots.

1. Para llevar a cabo esta comunicación se debe crear una cuenta en la página de Ubidots, dando clic en el siguiente link <https://ubidots.com/>.

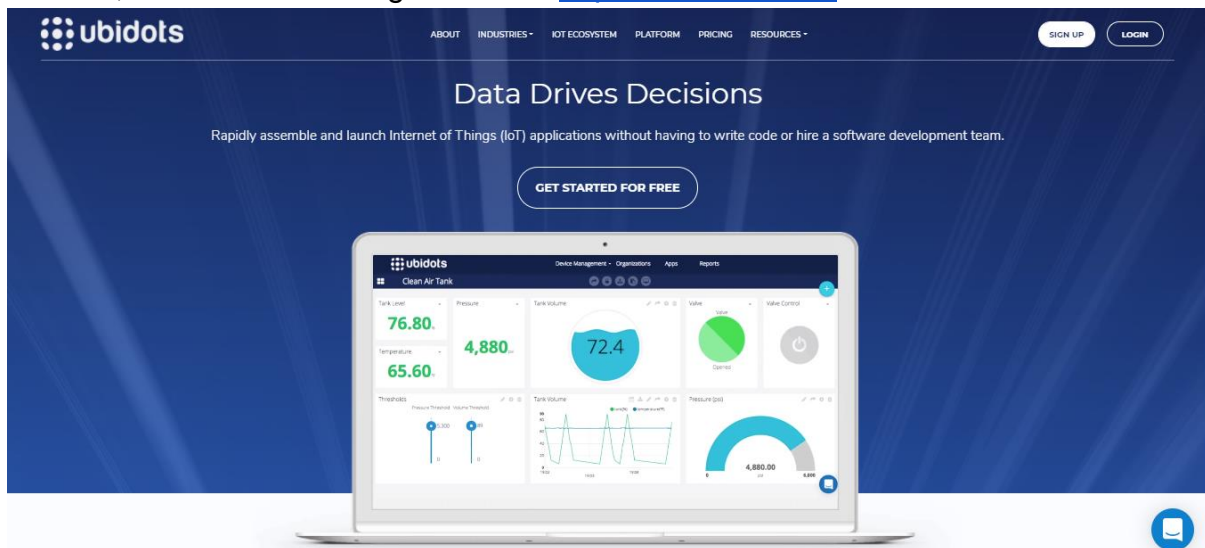


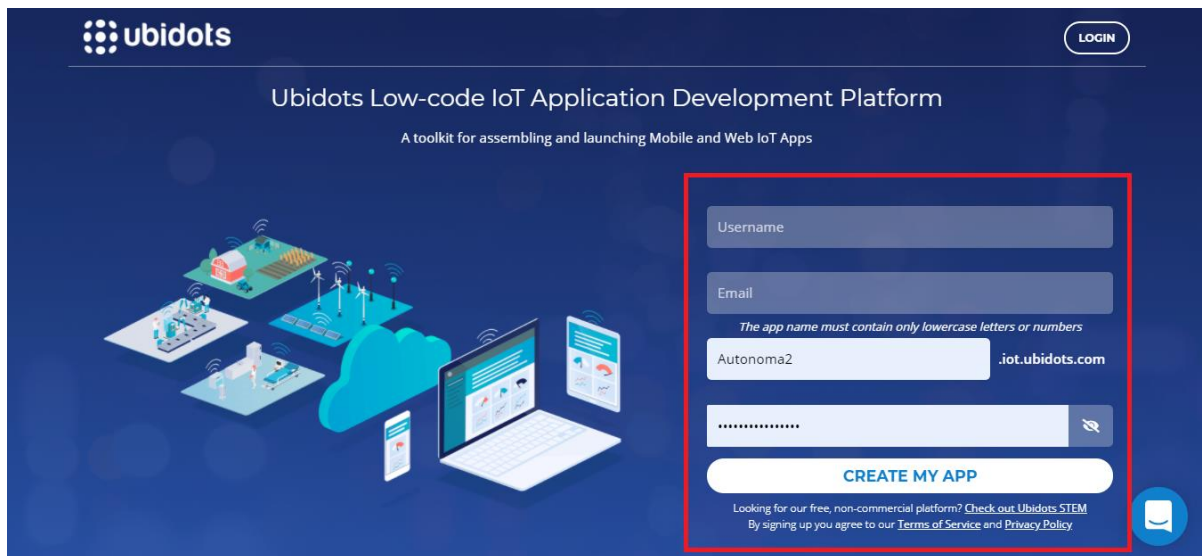
Ilustración 70. Configuración Ubidots - Web. Fuente Autor.

2. Se debe elegir la opción “Get Started for free” y posteriormente dar clic en “For Business” la cual ofrece 30 días de prueba gratuita y habilita todas los servicios por dicho tiempo.

The image is a screenshot of the Ubidots website's account configuration page. At the top left is the Ubidots logo. The navigation menu includes 'ABOUT', 'INDUSTRIES', 'IOT ECOSYSTEM', 'PLATFORM', 'PRICING', and 'RESOURCES'. On the top right are 'SIGN UP' and 'LOGIN' buttons. The main heading is 'Data Drives Decisions' with the subtext 'Rapidly assemble and launch Internet of Things (IoT) applications without having to write code or hire a software development team.' Below this is a 'GET STARTED FOR FREE' button, which is highlighted with a red box and a red arrow pointing to it from the right. In the center is a laptop displaying a dashboard for a 'Clean Air Tank' with various gauges and charts. Below the laptop, the text reads: 'Whether you're prototyping or going to market, Ubidots is designed to accelerate your IoT initiatives.' There are two main sections: 'For Educational or Personal Use' and 'For Business'. The 'For Educational or Personal Use' section features an illustration of a person with a rocket and a screen, with text: 'Join 60,000+ students, makers and professors using our FREE Ubidots STEM platform to prototype, learn, or teach IoT.' Below this is a 'TAKE ME TO UBIDOTS STEM' button. The 'For Business' section features an illustration of two people looking at a dashboard, with text: 'Join 1,000+ System Integrators, OEMs and IoT Entrepreneurs building connected products and services with Ubidots.' Below this is a 'CONTINUE →' button, which is highlighted with a red box and a red arrow pointing to it from the left.

Ilustración 71. Configuración Ubidots - Tipo de cuenta. Fuente Autor.

3. Llenar los campos requeridos.



The screenshot shows the Ubidots registration page. The header includes the Ubidots logo and a 'LOGIN' button. The main heading is 'Ubidots Low-code IoT Application Development Platform' with the subtitle 'A toolkit for assembling and launching Mobile and Web IoT Apps'. On the left, there is an illustration of IoT devices connected to a cloud. On the right, a registration form is highlighted with a red border. The form contains the following fields: 'Username', 'Email', 'App Name' (with a note: 'The app name must contain only lowercase letters or numbers'), and a password field. The 'App Name' field contains 'Autonoma2' and the domain '.iot.ubidots.com'. Below the fields is a 'CREATE MY APP' button. At the bottom of the form, there is a link for 'Looking for our free, non-commercial platform? Check out Ubidots STEM' and a note: 'By signing up you agree to our Terms of Service and Privacy Policy'.

Ilustración 72. Configuración Ubidots - Registro. Fuente Autor.

Configuración de Ubidots para la comunicación con la plataforma Sigfox

1. Dar clic en la opción device

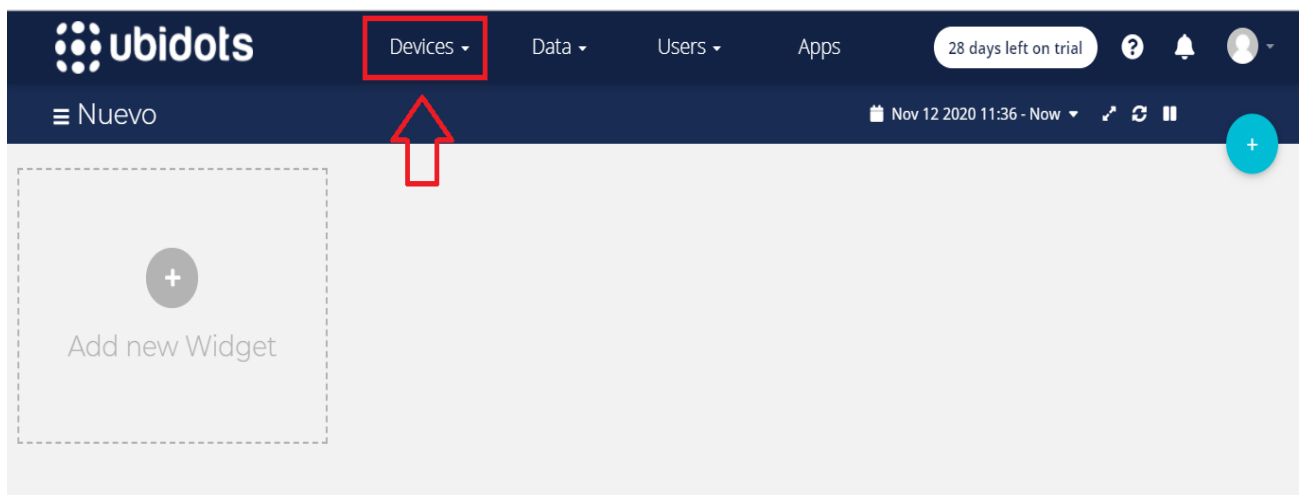


Ilustración 73. Configuración Ubidots - Devices. Fuente Autor.

2. Seleccionar el parámetro functions

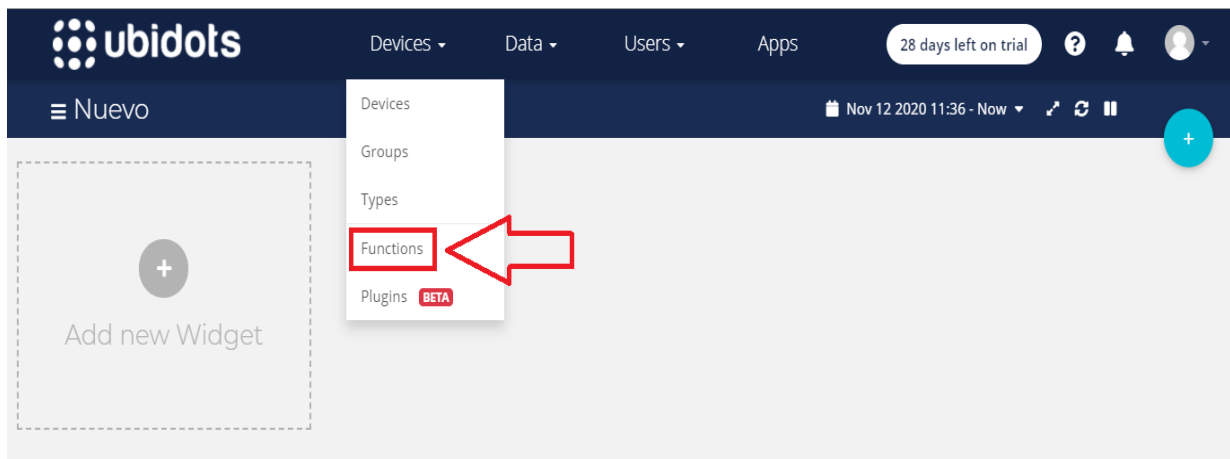


Ilustración 74. Configuración Ubidots - Functions. Fuente Autor.

3. Hacer clic en el icono con el signo “+” de la esquina superior derecha, para crear la función.

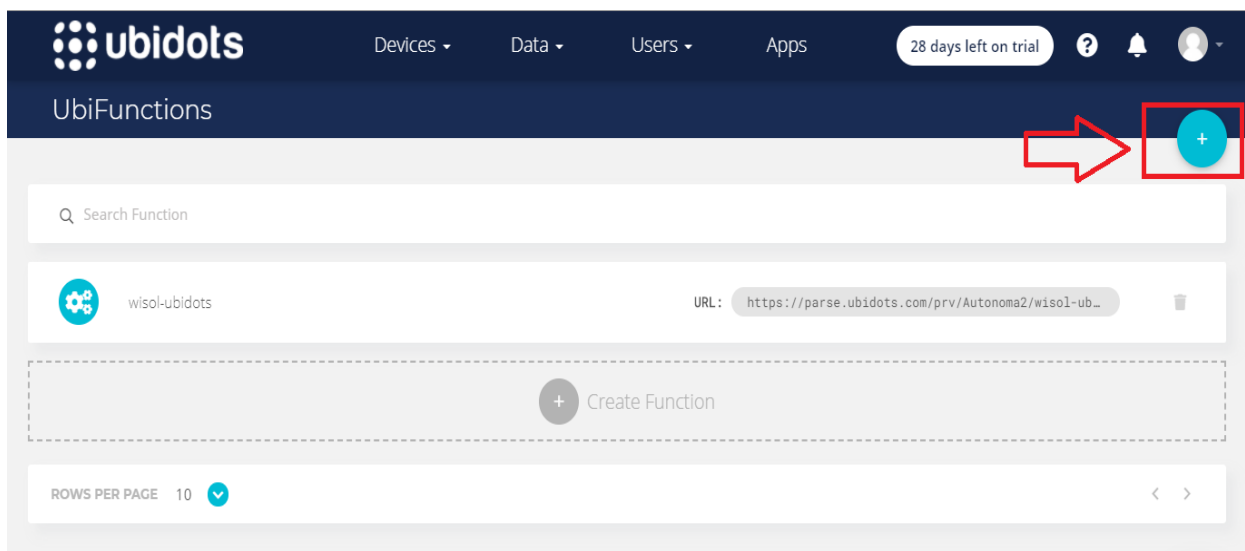


Ilustración 75. Configuración Ubidots - Nueva Functions. Fuente Autor.

4. Ingresar un **nombre** para la función, por ejemplo "Prueba_1", con el fin de identificarla en el backend de Sigfox más adelante. Para el **Método** HTTP, elegir la opción POST y seleccionar python 3.7 como método de ejecución.

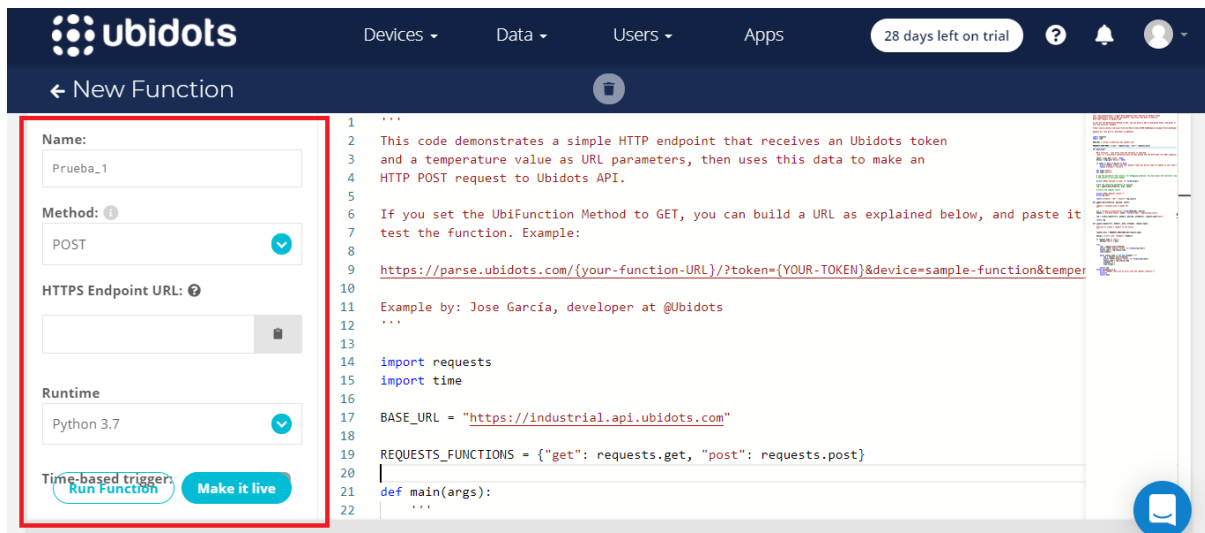


Ilustración 76. Configuración Ubidots - Configuración parámetros functions. Fuente Autor.

5. Elegir la opción "Make it live" esto para generar una URL de punto final de API que contenga el nombre de su función:

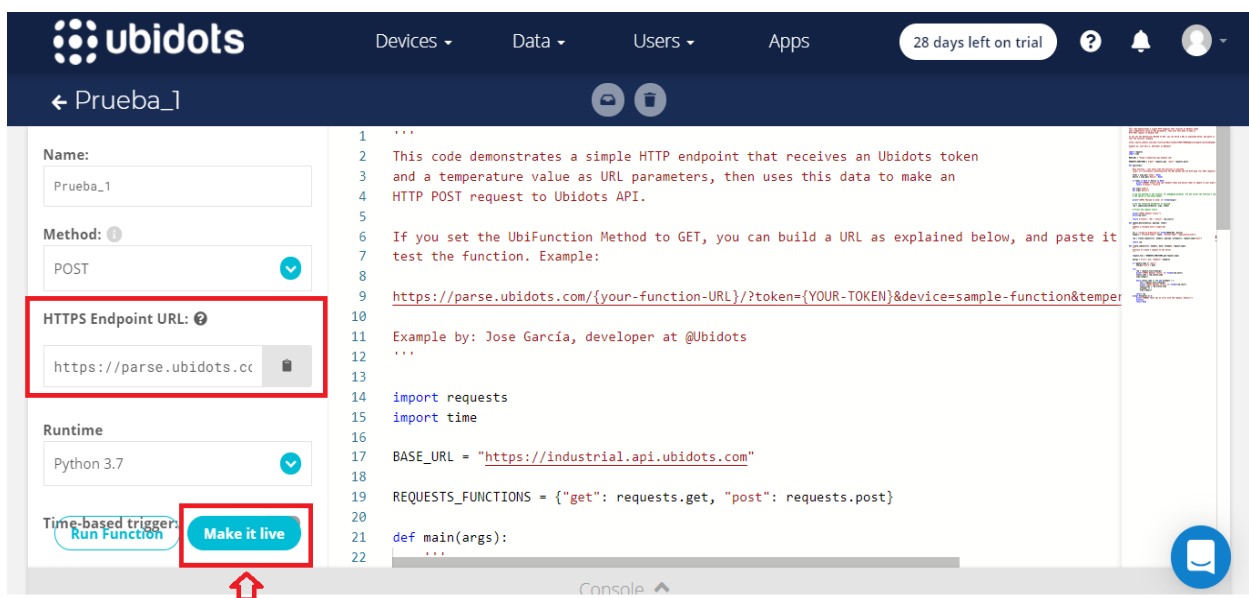


Ilustración 77. Configuración Ubidots - HTTPS Functions. Fuente Autor.

6. Se debe realizar la lógica de programación en lenguaje python para reconocer las diferentes tramas y así decodificarlas y finalmente poder visualizarlas en la plataforma.



Ilustración 78. Configuración Ubidots - Programación Functions. Fuente Autor.

7. Una vez se realice la lógica de programación es necesario asignar un TOKEN

Cada solicitud a Ubidots requiere un TOKEN. La forma más fácil de obtener la suya es hacer clic en "Credenciales de API" en el menú desplegable de usuarios.

-Vaya al menú desplegable de su usuario y haga clic en las credenciales de API:

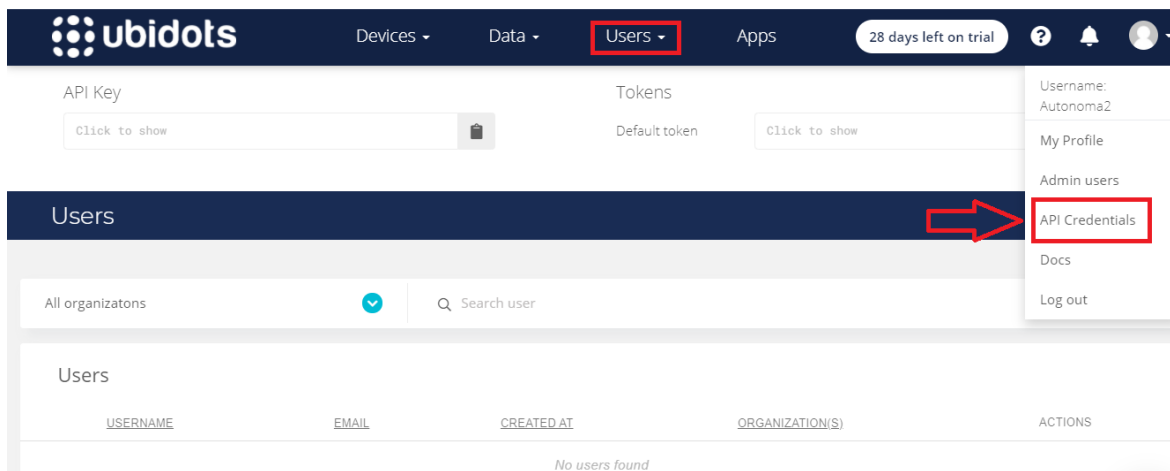


Ilustración 79. Configuración Ubidots - Api. Fuente Autor.

Habiendo realizado estos pasos la devolución de llamada de Sigfox está lista y UbiFunction está transformando correctamente los datos recibidos en la nube de Ubidots.

Comunicación Sigfox - Ubidots para la visualización de los datos

1. Luego de realizar el paso anterior, se procede a encender el sistema de monitoreo, donde se podrá visualizar la lectura de las tramas que llegan a Sigfox y posteriormente son enviadas mediante el callback a la plataforma Ubidots, estas tramas llegan decodificadas y legibles gracias a las funciones de Ubidots "UbiFunctions" para posteriormente ser analizadas.

The image displays two screenshots from the Sigfox and Ubidots platforms. The top screenshot shows the Sigfox 'Device 3FDB4E - Messages' page, which contains a table of received messages. The bottom screenshot shows the Ubidots 'Devices' page for device '3fdb4e', featuring a dashboard with various data widgets.

Time	Seq Num	Data / Decoding	LQI	Callbacks	Location
2020-11-11 21:45:08	853	01401d4a01c2992f80	High	Up	Location
2020-11-11 21:44:36	852	02428c428e428c428c428c	High	Up	Location
2020-11-11 21:43:39	851	01401d48f7c2992f80	Medium	Up	Location
2020-11-11 21:43:07	850	02428e428c428e428c428c	High	Up	Location

The Ubidots dashboard for device '3fdb4e' includes the following widgets:

- Latitude: 2.46
- Longitude: -76.59
- Position: 1.00
- tem_1: 22.58
- tem_2: 22.90
- tem_3: 22.58
- Latitude: 2.4576
- Longitude: 22.58

Ilustración 80. Comunicación Sigfox - Ubidots. Fuente Autor.

2. Para una mejor visualización de cada variable decodificada Ubidots permite crear tableros

Creación de tableros de visualización en Ubidots

1. Se hace clic en la opción data en la barra superior de la plataforma y luego se elige la opción “Dashboards”

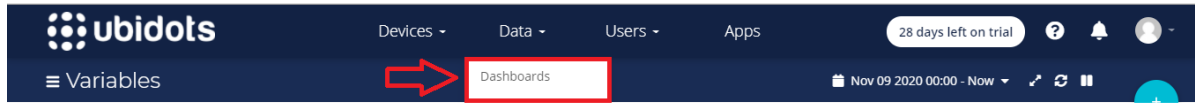


Ilustración 81. Configuración Ubidots - Dashboard. Fuente Autor.

2. Luego se busca la opción de las 3 barras juntas, posteriormente se debe elegir el botón con el símbolo “+”.

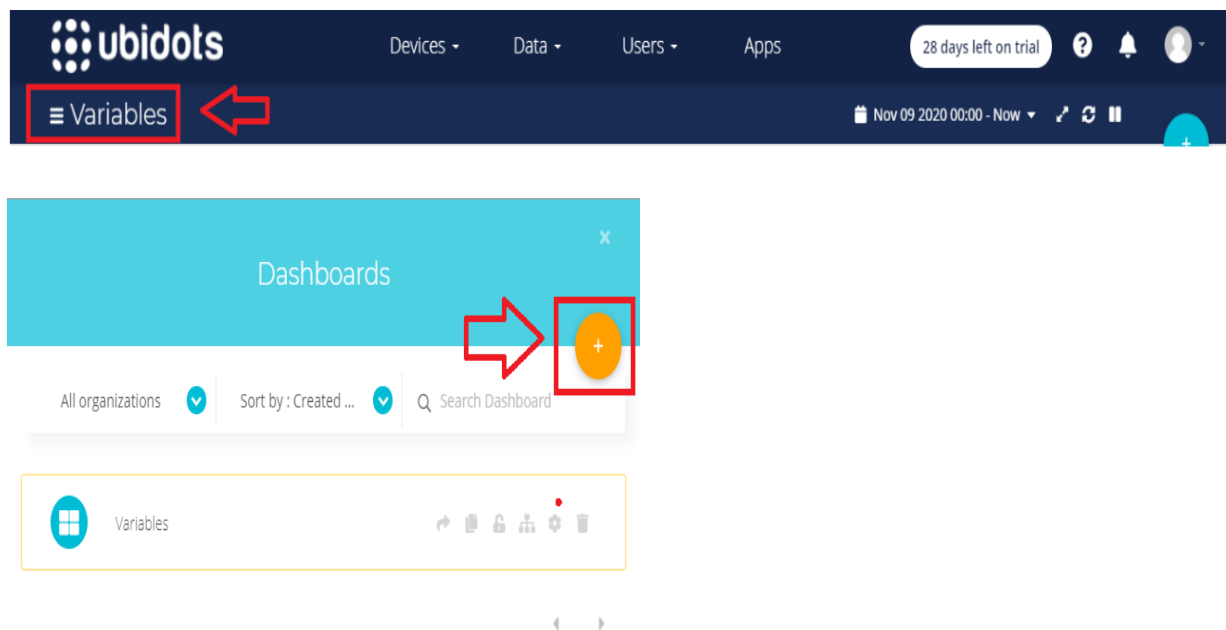


Ilustración 82. Configuración Ubidots - Nuevo Dashboard. Fuente Autor.

3. A continuación se llena el formulario del tablero.

Add new Dashboard ×

General Information

Name:

Default time range: ▼

Dynamic Dashboard:

Screen size: ▼

Date format: ▼

Appearance ▲

Floating widgets:

Widgets opacity:



 

Ilustración 83. Configuración Ubidots - Configuración Dashboard. Fuente Autor.

4. Ya creado el tablero se debe elegir el widget de acuerdo al dato que se quiere visualizar.

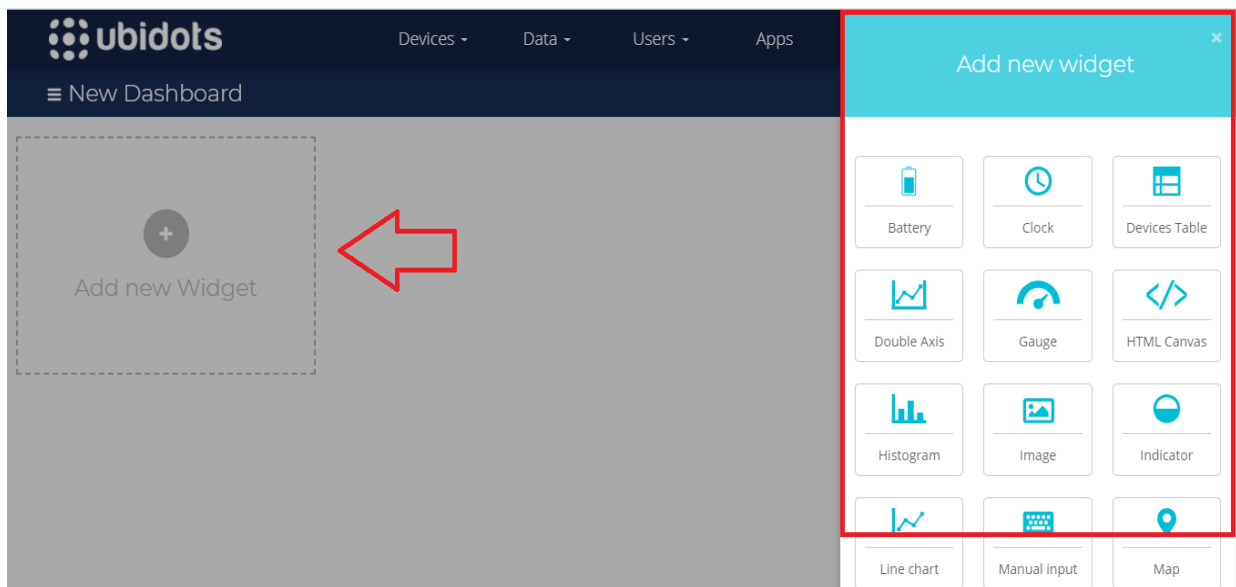


Ilustración 84. Configuración Ubidots - Widgets Dashboard. Fuente Autor.

5. Elegido el widget, se debe llenar la información con la variable que se necesite mostrar y así mismo con los parámetros que se requieran.

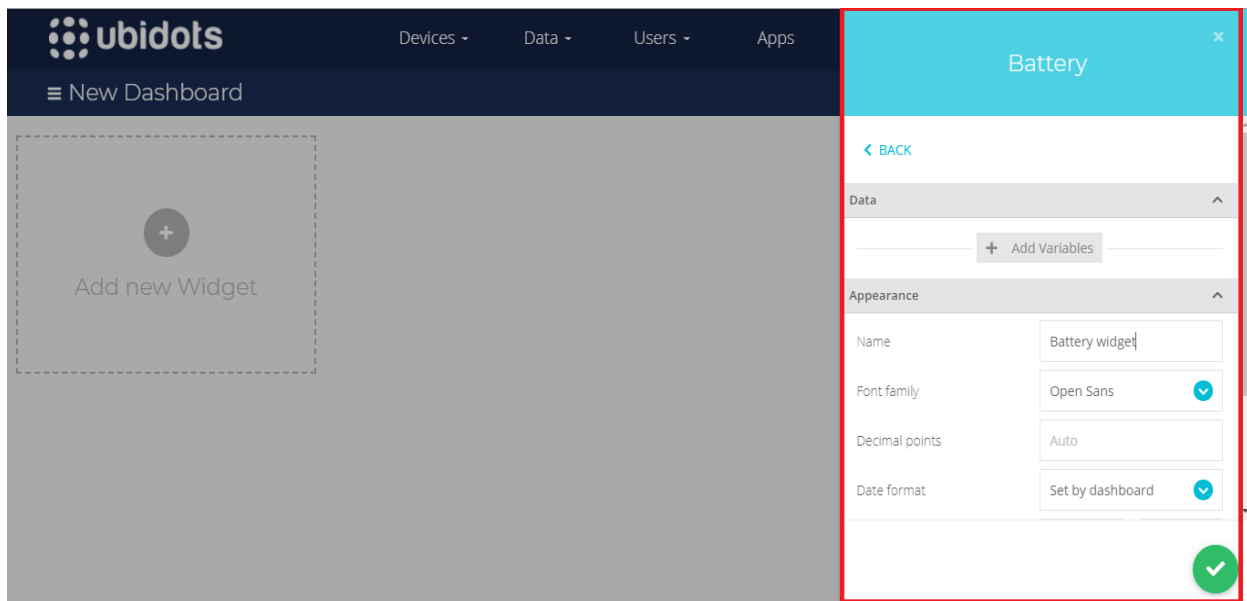


Ilustración 85. Configuración Ubidots - Ajuste Widget Dashboard. Fuente Autor.

6. Finalmente se puede observar todas las variables con su respectivo widget en el tablero de Ubidots.

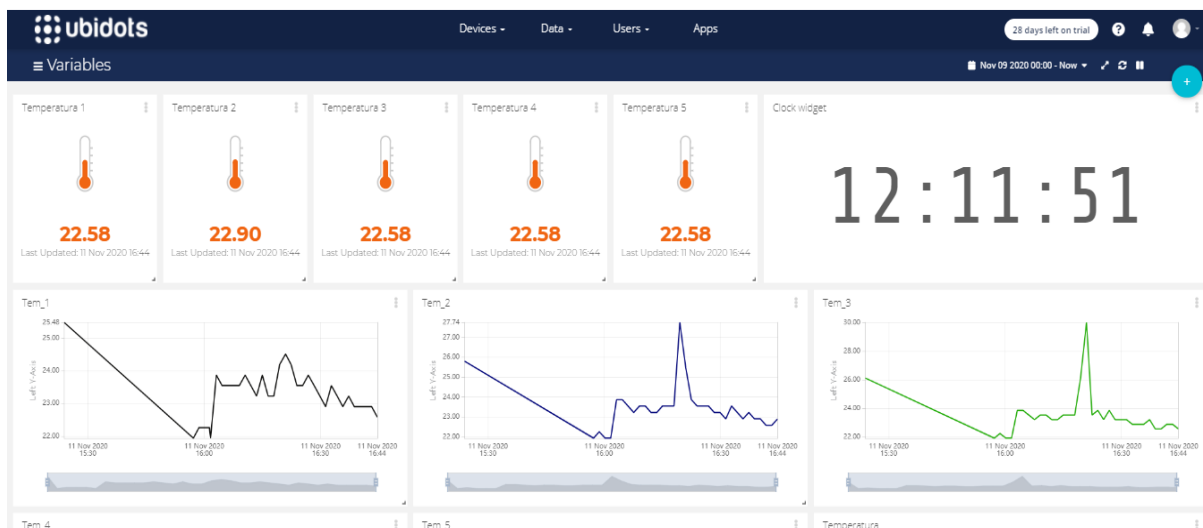


Ilustración 86. Configuración Ubidots - Visualización Variables. Fuente Autor.

Anexo C. Pruebas en campo

En este apartado se presentan las fotos tomadas en las pruebas de campo.



Ilustración 87. Finca elegida para realizar las pruebas de campo. Fuente Autor.



Ilustración 88. Selección del bovino para realizar las pruebas. Fuente Autor.



Ilustración 89. Signos fisiológicos de celo - 19 Septiembre. Fuente Autor.



Ilustración 90. Signos Fisiológicos del celo - 19 de Septiembre. Fuente Autor.