

# **IMPLEMENTACIÓN DE UNA RED DE SENsoRES INALÁMBRICOS BASADA EN LPWAN PARA LA GEORREFERENCIACIÓN DE PUNTOS DE CONTROL TERRESTRES**



**UNIVERSIDAD DISTRITAL  
FRANCISCO JOSÉ DE CALDAS**

PROPONENTES  
**Andrés Camilo González Hernández**

DIRECTOR  
**PhD. Elvis Eduardo Gaona García**

CODIRECTOR  
**MSc. Víctor Daniel Angulo Morales**

Facultad de Ingeniería  
Ingeniería Electrónica  
Universidad Distrital Francisco José de Caldas  
Bogotá  
2020.



# ÍNDICE GENERAL

<b>Índice de figuras</b>	<b>v</b>
<b>Índice de cuadros</b>	<b>vii</b>
<b>Glosario</b>	<b>ix</b>
<b>1 Introducción</b>	<b>1</b>
1.1 Problema de Investigación . . . . .	3
1.2 Pregunta de Investigación. . . . .	3
1.3 Justificación. . . . .	5
1.4 Objetivo General . . . . .	6
1.5 Objetivos Específicos . . . . .	6
<b>2 Marco De Referencia</b>	<b>7</b>
<b>3 Marco Teórico</b>	<b>9</b>
3.1 Sistema Global De Navegación Por Satélite . . . . .	9
3.1.1 Fuentes De Error. . . . .	11
3.2 Redes de área amplia y baja potencia (LPWAN) . . . . .	12
3.2.1 Tecnologías LPWAN . . . . .	13
3.2.2 Comparación en términos de factores IoT . . . . .	18
3.2.3 Control de Acceso al medio . . . . .	19
3.3 Computadoras de Placa reducida . . . . .	22
3.4 Topologías de Red . . . . .	24
3.4.1 Estrella. . . . .	25
3.4.2 Malla. . . . .	26
3.4.3 Híbridas . . . . .	27
<b>4 Metodología</b>	<b>29</b>
4.1 Análisis de requerimientos . . . . .	29
4.2 Diseño e implementación de la Red. . . . .	30
4.3 Desarrollo de Interfaz Gráfica . . . . .	30
4.4 Pruebas de Campo . . . . .	30
<b>5 Desarrollo</b>	<b>31</b>
5.1 Análisis y Requerimientos. . . . .	31
5.1.1 Dispositivo para el posicionamiento . . . . .	31
5.1.2 Dispositivo para el procesamiento . . . . .	33

5.2 Dispositivo de Comunicaciones . . . . .	34
5.2.1 Calidad del Servicio . . . . .	34
5.2.2 Latencia . . . . .	34
5.2.3 Cobertura de red y alcance. . . . .	34
5.2.4 Modelo de despliegue y disponibilidad . . . . .	34
5.2.5 Escalabilidad y longitud de carga útil . . . . .	35
5.2.6 Dispositivo Seleccionado . . . . .	36
5.3 Diseño e implementación de la Red. . . . .	38
5.3.1 Arquitectura de red . . . . .	38
5.3.2 Implementación Red. . . . .	39
5.4 Diseño de Interfaz Gráfica . . . . .	41
5.4.1 Modelos . . . . .	41
5.4.2 Vistas . . . . .	42
5.4.3 Templete . . . . .	42
5.4.4 Lenguaje Programación y Framework . . . . .	45
5.4.5 Implementación . . . . .	46
5.5 Pruebas de Campo . . . . .	49
5.5.1 LPWAN . . . . .	49
5.5.2 GNSS. . . . .	49
5.5.3 Red de Sensores . . . . .	54
<b>6 Conclusiones</b>	<b>59</b>
<b>7 Trabajos Futuros</b>	<b>61</b>
<b>Bibliografía</b>	<b>63</b>
<b>A Apéndices</b>	<b>67</b>
A.1 Configuración De Raspbian . . . . .	67
A.2 Configuración De U-blox zed-F9P . . . . .	71
A.3 Manual Usuario . . . . .	75
A.4 Código configuración de nodos . . . . .	77
A.5 Código Lectura De GNSS . . . . .	81
A.6 Repositorio Github . . . . .	83

# ÍNDICE DE FIGURAS

3.1 Representación esquemática del funcionamiento del GNSS . . . . .	10
3.2 Velocidad de trasferencia de datos vs alcance para diferentes tecnologías	12
3.3 Arquitectura de la red LoRa . . . . .	14
3.4 Espectrograma de señal LoRa mediante lectura SDR . . . . .	15
3.5 Red Configurada en estrella . . . . .	25
3.6 Red Configurada en Malla . . . . .	26
3.7 Red Híbrida . . . . .	27
5.1 C099-F9P Application Board . . . . .	32
5.2 Raspberry Pi Zero Wireless . . . . .	33
5.3 Comparación entre Sigfox, LoRa y NB-IoT en términos de factores de IoT	35
5.4 Semtech SX1272 LoRa Mbed Enabled Shield . . . . .	37
5.5 Conexiones de un Nodo de la red . . . . .	38
5.6 Topología de la red de comunicaciones . . . . .	39
5.7 Nodo de la Red (Raspberry pi Zero + sx1272 + Bridge) . . . . .	40
5.8 Modelo Final de la red Con nodos Reales . . . . .	40
5.9 Modelo Funcionamiento django Fuente Platzi . . . . .	41
5.10 Vistas Entorno Gráfico : Wi fi-Login . . . . .	43
5.11 Vistas Entorno Gráfico :Lista de Nodos . . . . .	44
5.12 Vistas Entorno Gráfico :Detalles de Nodos . . . . .	44
5.13 Comparación Python,PHP,Node.js server response . . . . .	45
5.14 Comparación Python,PHP,Node.js Serie de Fibonacci . . . . .	46
5.15 Vistas Django (Inicio) . . . . .	47
5.16 Vistas Django (Detalles) . . . . .	48
5.17 Tiempos de envío promedio con SF 9 . . . . .	50
5.18 Tiempos de envío promedio con SF 7 . . . . .	50
5.19 Paquetes Perdidos SF 9 . . . . .	51
5.20 Paquetes Perdidos SF 7 . . . . .	51
5.21 Posicionamiento de puntos . . . . .	52
5.22 Desviación En X . . . . .	53
5.23 Desviación En Y . . . . .	53
5.24 Desviación En Z . . . . .	53
5.25 Vista real del nodo Implementado . . . . .	55
5.26 Experimento 1 . . . . .	55
5.27 Experimento - Respuesta de la red con SF de 7 en 200 m y 400 m . . . . .	57
5.28 Experimento - Respuesta de la red con SF de 9 en 200 m y 400 m . . . . .	58
5.29 Experimento - Paquetes Perdidos . . . . .	58

A.1	Interfaz de SD Card Formatter . . . . .	68
A.2	Interfaz Etcher . . . . .	69
A.3	Interfaz PuTTy . . . . .	69
A.4	Raspi-Config . . . . .	70
A.5	Administrador de Dispositivos . . . . .	71
A.6	Seleccionar puerto U-Center . . . . .	72
A.7	U-Center . . . . .	72
A.8	Generation 9 Configuration view . . . . .	73
A.9	Message View . . . . .	74
A.10	Configure View . . . . .	74
A.11	Vistas Django (Inicio sección) . . . . .	75
A.12	Vistas Django (Lista de Nodos) . . . . .	76
A.13	Vistas Django (Detalles) . . . . .	76

# ÍNDICE DE CUADROS

3.1 Visión general de las tecnologías LPWAN: Sigfox, LoRa y NB-IoT . . . . .	19
3.2 Comparación de placas Raspberry . . . . .	23
5.1 ZED-F9P Rendimiento en diferentes modos GNSS . . . . .	32
5.2 Modelo De lista de Archivos . . . . .	41
5.3 Modelo Maestro-Esclavo . . . . .	41
5.4 Modelo De Nodos . . . . .	42
5.5 Resultados con distancia 200 m . . . . .	56
5.6 Resultados con distancia 400 m . . . . .	56



# GLOSARIO

**GLONASS** Global Navigation Satellite System.

**GNSS** Global Navigation Satellite System.

**GPS** Global Positioning System.

**IoT** Internet de las cosas.

**IQRF** IQRF.

**IRNSS** Indian Regional Navigation Satellite System.

**LoRa** Long range.

**LPWAN** Low-Power Wide-Area Network.

**M2M** Máquina a Máquina.

**MAC** Media Access Control.

**QZSS** Sistema Satelital Cuasi-Zenith.

**RAW(low-level data)** low-level data.

**RF** Radiofrecuencia.

**RSSI** Indicador de Intensidad de Señal Recibida.

**RTK** Real Time Kinematic.

**SBC** Computadoras de placa reducida.

**SF-PPP** Posicionamiento de punto preciso de frecuencia única.

**SigFox** SigFox.

**WSN** Red de Sensores Inalámbricos (Wireless Sensor Network).



# 1

## INTRODUCCIÓN

Las redes de sensores inalámbricas son formados por dispositivos autónomos, utilizados para el monitoreo sobre un espacio de variables físicas y ambientales, un red se base en módulos que son fácilmente manipulables que permiten a las redes tener crecimientos y modificaciones en su funcionamiento.

Las redes de sensores se caracterizan por ser redes desatendidas, habitualmente construidas de manera descentralizada que no depende de una infraestructura pre-existente para resolver un problema muy específico . Las comunicaciones inalámbricas son una solución eficaz y confiable en la automatización del hogar y la oficina, en tal sentido, varios medios de transmisión de señales podrían utilizarse, incluyendo la luz y el ultrasonido, pero haciendo una comparación entre ellos con respecto al posible tráfico de datos, los precios y el área de cobertura, terminan finalmente siendo más atractivos los enlaces de radio frecuencia Radiofrecuencia (RF)

Actualmente el uso de Red de Sensores Inalámbricos (Wireless Sensor Network) (WSN) a presentado un incremento notable mientras debido a que estos sistemas permiten capturar información del área geográfica en la que son desplegados. Como consecuencia, las aplicaciones de las WSN van desde el desarrollo de sistemas inteligentes para el transporte público , hasta la monitorización de variables ambientales para desarrollar sistemas agrícolas inteligentes.

Uno de los problemas que se presentan en la actualidad es el levantamiento topográfico de terrenos la cual se utilizan par la reconstrucción digital de terrenos, para la realizar un correcto levantamiento topográfico tradicional por medio de Global Navigation Satellite System (GNSS) de precisión se requieren tiempo prolongados de medición, así como tener una visual entre los puntos que se quieran tomar, un ejemplo de esto se presentó en un estudio de la universidad politécnica de cartagena (España) en la cual tardaron 3 días con aproximadamente 6 horas de trabajo diario para establecer los diferentes puntos requeridos para realizar el levantamiento([Simón Toquero et al., 2011](#)).

Para lo cual se implementara una red de área amplia de baja potencia (LPWAN) tomando en cuenta que se realizara una análisis de requerimientos entre dos de sus principales modelos que son LoRaWAN y SigFox , como también diferentes dispositivos sistema global de navegación por satélite (GNSS) con doble constelación preferiblemente GPS/GLONASS y con frecuencia sencilla o doble para realizar la georreferenciación de los puntos y por último se buscara un Computadoras de placa reducida (SBC) que se encargara del procesamiento de los datos.

En el documento se ilustra en primer lugar un marco de referencia sobre proyectos similares realizados,seguidos de un marco teórico sobre los diversos conceptos que se utilizaron, seguido de la metodología y desarrollado del proyecto, por ultimo resultados ,conclusiones y proyectos futuros.

## 1.1. PROBLEMA DE INVESTIGACIÓN

En la actualidad, cada vez es más amplio el catálogo de funciones y aplicaciones de los vehículos aéreos no tripulados o drones. La mayoría de estos usos se realizan en espacios abiertos, donde pueden hacer uso de un sistema de posicionamiento global como el GPS.

La precisión de los GPS son de varios metros (incluso 10 – 20 m) (Force, 2018), por lo que aunque el uso de drones permiten hacer un reconocimiento mucho más rápido sobre un terreno, se requiere que estén georreferenciadas las imágenes para poder hacer una reconstrucción más precisa como lo muestra Rubio Sabina en la reconstrucción de castillos medievales del Campo de Montiel (Sabina et al., 2015) donde por medio de fotografías aéreas se reconstruye el terreno pero aun así se requiere realizar la topografía básica para obtener los puntos y realizar con mayor precisión la reconstrucción del terreno.

Para lo cual se plantea implementar una red de puntos de control que permitan obtener datos más rápido al momento de hacer la reconstrucción sobre un terreno, estos puntos de control se deberían poder repartir homogéneamente sobre el terreno de estudio, para obtener los datos, obteniendo así reconstrucciones del terreno más reales.

Para esto se implementaría una red de área amplia de baja potencia (LPWAN) tomando en cuenta que se realizaría una análisis de requerimientos entre dos de sus principales modelos que son LoRaWAN y SigFox, como también diferentes dispositivos sistema global de navegación por satélite (GNSS) con doble constelación preferiblemente GPS/GLONASS y con frecuencia sencilla o doble para realizar la georreferenciación de los puntos y por último se buscaría un SBC que se encargara del procesamiento de los datos.

Entre las diferentes redes se encuentran las redes 3G y 4G pero para el caso en de estudio el consumo energético y el equipamiento necesario para la implementación presenta costos elevados, en contraste redes implementadas con un cobertura muy pequeña como lo son ZigBee o bluetooth, no tiene grandes áreas de cobertura, por último se encuentran las redes basadas en Low-Power Wide-Area Network (LPWAN) que poseen una gran cobertura con bajo consumo de energía y baja tasa de bits.

## 1.2. PREGUNTA DE INVESTIGACIÓN

En este orden de ideas, se generan las siguientes preguntas de investigación que define el problema por solucionar:

¿Es posible implementar una red de baja potencia (LPWAN) para la georeferenciación de puntos de control terrestre sobre un terreno con la capacidad de obtener los datos desde cualquier punto de la red?

Esta pregunta de Investigación genera una hipótesis:

Es posible implementar una red basada en LPWAN que permita al usuario obtener datos de toda la red, desde cualquier punto de esta teniendo en cuenta que la red no está

conectada a una red en la nube ya que se plantea el uso en entornos rurales donde no se espera tener acceso a puntos de Ethernet.

### 1.3. JUSTIFICACIÓN

Al implementar una red de sensores que se encargara de la georreferenciación no se necesitaría una intervención directa sobre el terreno, en caso de ser zonas protegidas por su flora y fauna, o minas la intervención sería lo mínimo posible sobre el terreno, evitando posibles lesiones a personas o daños en la propiedad, como también disminuiría los tiempos de operación para tomar puntos requeridos sobre el terreno.

Se economizarían costos al no requerir tanta mano de obra para la toma de puntos de control, como también se reducirían costos en los equipos necesarios para generar los puntos de control de forma tradicional.

A lo largo de los años han aparecido muchas formas con las cuales se puede controlar el desplazamiento y posicionamiento de un dron, estas hacen uso de todo tipo de sensores, siendo los más populares el GNSS y los sensores iniciales (IMU). Sin embargo, aunque es posible llegar a tener un buen control, estos sistemas no se han enfocado hacia la precisión de posicionamiento en un punto específico del planeta. Con ayuda de sistemas actuales y corrección de errores con cálculos más potentes se puede hacer que este error de posicionamiento sea muy pequeño.

La redes inalámbricas son ampliamente utilizadas para la monitorear cultivos como se utilizan en ([Urbano-Molano, 2013](#)) la red que se implementa permite obtener las variables ambientales en cultivos de café colombiano.

Este trabajo se desarrollará con el fin profundizar en uno de los objetivos del trabajo de investigación Evaluación de una red IoT de rango amplio bajo el estándar LoRa" que fue financiado por el CIDC, enfocándose principalmente el objetivo relacionado con de analizar y evaluar una red implementada para el propósito de validación y seguimiento. En cuanto a cobertura, rata de bits y latencia de la red.

## 1.4. OBJETIVO GENERAL

Diseñar e implementar un red de sensores inalámbrica de sistemas de posicionamiento global basada en LPWAN para georeferenciar puntos de control terrestres.

## 1.5. OBJETIVOS ESPECÍFICOS

- Realizar un análisis de requerimientos para receptores de posicionamiento global con capacidad de entregar información en formato low-level data (RAW(low-level data)) así como los dispositivos de la red.
- Diseñar e implementar una red basada en LPWAN para la trasferencia de información que permita obtener los datos desde cualquier punto de la red.
- Desarrollar el software que permita por medio de puntos de acceso visualizar los datos del sensor ( posición ) y de los parámetros de la red tales como latencia, tiempos de espera,etc.
- Realizar las pruebas de campo que permitan verificar la funcionalidad de la red de sensores.

# 2

## MARCO DE REFERENCIA

Los sistemas global de navegación por satélite (GNSS) son ampliamente utilizados en todo tipo de aplicaciones en bases a las diversas constelaciones disponibles entre las cuales encontramos Global Positioning System (GPS), Global Navigation Satellite System (GLONASS) ,GALILEO,Sistema Satelital Cuasi-Zenith (QZSS) e Indian Regional Navigation Satellite System (IRNSS) como se ilustro en el artículo presentado en el COSPAR 2015 por O. Montenbruck ,donde se realizó una comparación entre las diferentes constelaciones GNSS, revisando su precisión y los desfases respecto a un punto de referencia ([Montenbruck et al., 2015](#)).

En la Universidad de Delft, Países Bajos, se desarrolló un algoritmo de posicionamiento denominado Posicionamiento de punto preciso de frecuencia única (SF-PPP). Utilizando un receptor de bajo costo con frecuencia única, antena sencilla y única constelación GNSS (GPS). El receptor proporciona mediciones de RAW(low-level data) a la SF-PPP que se implementó Raspberry Pi versión 2, obteniendo errores cercanos a 0.5 m manteniendo un bajo costo ([Rademakers et al., 2016](#)).

En la universidad de Antwerp (Bélgica) se realizó una comparación entre Long range (LoRa) y SigFox (SigFox) utilizando el método de Indicador de Intensidad de Señal Recibida (RSSI) con fingerprinting en una zona urbana y rural, en donde se obtuvo un error menor en la implementación en la zona urbana con LoRa en comparación con Sigfox ([Aernouts et al., 2018](#)).

En el “IEEE transactions on geoscience and remote sensing” de 2015 se presentó un artículo por Mieczysław Bakula donde se realizaron pruebas con GPS/GLONASS bajo condiciones de observación difíciles en el “Kortowo Park”, Las pruebas prácticas para la tecnología presentada mostraron que con tres receptores GPS / GLONASS móviles se podía obtener bastante eficiente en entornos forestales (a diferencia del Real Time Kinematic (RTK)) y permitían una precisión centimétrica confiable de las posiciones con secciones de 5 minutos de observación ([Bakula et al., 2014](#)).

Las LPWAN son ampliamente utilizadas en aplicaciones Máquina a Máquina (M2M) y de Internet de las cosas (IoT). En la actualidad, ambos entornos tienen limitaciones

tanto por la inversión necesaria como también por cuestiones relacionadas con la distancia de la fuente de energía. Las redes LPWAN parecen solventar estos problemas al ofrecer bajo consumo de energía, coste reducido y largo alcance, estas redes hacen posible tener muchos miles de sensores recopilando y enviando datos a menor costo, a más largo alcance, y con mejor duración de la batería que otras opciones de conectividad entre sus aplicaciones encontramos :

En la universidad de Praga (República Checa) en se presentó un trabajo por el investigador Lukas Krupka sobre LPWAN, donde se describen los términos de LPWAN y sus representantes LoRa, SigFox e IQRF (IQRF), donde se inspeccionaron características , topologías y algunas técnicas de implementan, de donde LoRa y SigFox prometen un gran desarrollo y más de miles de dispositivos colocados con un la capacidad de cubrir una gran área ([Krupka et al., 2016](#)).

Siendo LoRa una de las alternativas de implementan para una LPWAN en Brasil en el 2017 se implementó una red de módulos LoRa para el monitoreo de navegación, El sistema tenía una baja tasa de pérdida de paquetes por debajo del 6 % y un largo alcance de más de 2 km en las zonas planas pero se ve influenciada por obstáculos tales como edificios altos y árboles, que conducen a la alta tasa de pérdida de paquetes de más del 34 % en aquellas zonas con muchos obstáculos ([Li et al., 2017](#)).

Los minicomputadores Raspberry pi son ampliamente utilizados en todo tipo de aplicaciones por su versatilidad en los lenguajes de programación que soporta como también a su gran capacidad de procesamiento de información, en Ecuador - Guayaquil en la Escuela Superior Politécnica del Litoral (ESPOL) se presentó un trabajo en el que se realizaba el control de velocidad y posicionamiento de un automóvil mediante el uso del minicomputador moderno Raspberry Pi programado en Python , el cual está provisto de un módulo GPS de la empresa Vicotech y acelerómetro para poder establecer la posición del vehículo, entre los problemas que encontraron fue que la precisión dependía de la caridad de satélites a los que se conectara el modulo gps ([Castrillón, 2013](#)).

En la universidad Distrital se emplean módulos LoRa para la comunicación entre nodos generando una red de sensores sobre dos ríos para monitoreo de la cálida del agua (CE, pH, TDS, T), entre los problemas encontrados surge por acople de impedancias ,como también la necesidad de antenas de alta potencia para cubrir el terreno.([Ordoñez et al., 2017](#)),

# 3

## MARCO TEÓRICO

### 3.1. SISTEMA GLOBAL DE NAVEGACIÓN POR SATÉLITE

Los sistemas de posicionamiento global por satélite o GNSS por sus siglas en inglés, son sistemas formados por constelaciones de satélites que proveen servicios de posicionamiento, navegación y sincronización en una región o a nivel global. Este tipo de tecnología fueron inicialmente desarrolladas para propósitos militares. Sin embargo con el fin de mantener el paradigma de movilidad, estas tecnologías deben buscar complementarse.

Actualmente se consideran que hay tres servicios funcionando de manera global. GPS fue el primer sistema en funcionar y es propiedad del gobierno de Estados Unidos, contando actualmente con 29 satélites. GLONASS es el sistema operado por Rusia, contando con 24 satélites y por último Galileo este es el sistema creado por la Unión europea, el cual espera tener en funcionamiento más de 24 satélites para 2020. En cuanto a GNSS de área Regional, se encuentran. QZSS el cual es un sistema operado por el gobierno de Japón teniendo una constelación de 4 satélites en 2018 y planeando tener 7 satélites para 2023 donde una de sus funciones será ampliar la cobertura del sistema GPS en el este de Asia y Oceanía. IRNSS es un sistema operado por el gobierno de la India, el cual posee 7 satélites operativos. Por último está Beidou, es el sistema creado y operado por el gobierno de China, el cual planea tener 35 satélites operativos para 2020. ([Accuracy, 2013](#))

Los sistemas de navegación por satélite tienen una estructura claramente definida, que se divide en tres segmentos distintos: un segmento espacial, un segmento de control, un segmento de usuarios. No se entiende un GNSS sin alguno de estos tres elementos.

El funcionamiento de un Sistema de Navegación por Satélite involucra los distintos segmentos, de tal manera que se relacionan entre sí:

- **Segmento espacial:** envía la señal que se recibe en los segmentos de control y usuario.

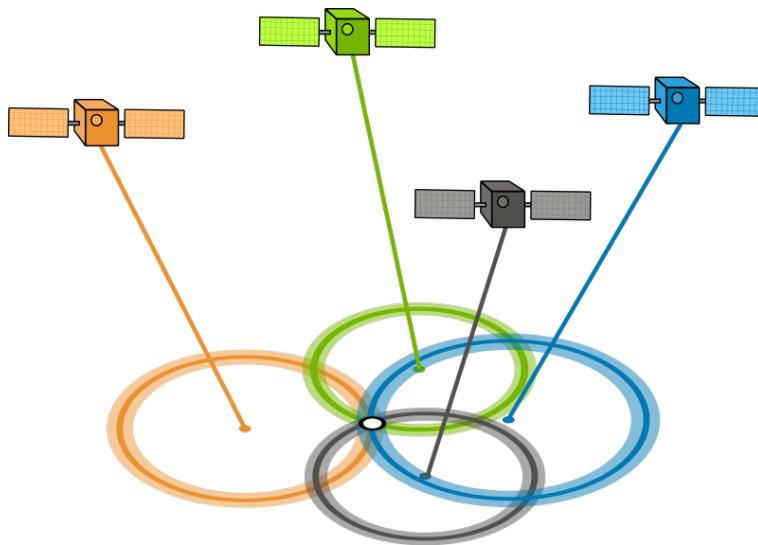


Figura 3.1: Representación esquemática del funcionamiento del GNSS  
 Fuente: ([Álvarez, 2008](#))

- **Segmento de control:** recibe la señal del segmento de espacio, monitoriza y actualiza información enviando correcciones a los satélites si es preciso.
- **Segmento de usuario:** recibe información procedente del segmento espacial y calcula su posición.

El funcionamiento del sistema GPS se basa en el principio matemático de la triangulación. Por tanto, para calcular la posición de un punto será necesario que el receptor GNSS determine con exactitud la distancia que lo separa de los satélites.

Como se explicó anteriormente, con la aplicación del principio matemático de la triangulación podemos conocer el punto o lugar donde nos encontramos situados, e incluso rastrear y ubicar el origen de una transmisión por ondas de radio. El sistema GNSS utiliza el mismo principio, pero en lugar de emplear círculos o líneas rectas crea esferas virtuales o imaginarias para lograr el mismo objetivo.

Desde el mismo momento que el receptor GPS detecta una señal de radiofrecuencia transmitida por un satélite desde su órbita, se genera una esfera virtual o imaginaria que envuelve al satélite. El propio satélite actuará como centro de la esfera cuya superficie se extenderá hasta el punto o lugar donde se encuentre situada la antena del receptor; por tanto, el radio de la esfera será igual a la distancia que separa al satélite del receptor. A partir de ese instante el receptor GNSS medirá las distancias que lo separan como mínimo de dos satélites 3.1. Para ello tendrá que calcular el tiempo que demora cada señal en viajar desde los satélites hasta el punto donde éste se encuentra situado y realizar los correspondientes cálculos matemáticos ([Álvarez, 2008](#)).

### 3.1.1. FUENTES DE ERROR

La información procedente de cada satélite que viaja en una señal puede verse afectada por distintas fuentes de error:

- **Efectos atmosféricos:** La troposfera y la ionosfera pueden cambiar la velocidad de propagación de una señal GPS. Debido a las condiciones atmosféricas, la atmósfera refracta las señales de los satélites cuando las atraviesan en su camino hacia la superficie terrestre.
- **Efectos multirayecto :** Una posible fuente de error en los cálculos del GPS es el efecto multirayecto. El multirayecto se produce cuando la señal del satélite GPS rebota en estructuras cercanas como edificios y montañas.
- **Errores de efemérides y reloj:** Utilizando la Trilateración de señales de tiempo en órbita, los receptores GPS en tierra pueden obtener posiciones precisas. Pero debido a la inexactitud del reloj atómico del satélite que es sincronizado, esto puede compensar una medida de posición por 2 metros más o menos.

Estos errores que degradan la precisión del servicio, han hecho que surjan como ya se han comentado sistemas de mejora para obtener mayores prestaciones.

### 3.2. REDES DE ÁREA AMPLIA Y BAJA POTENCIA (LPWAN)

Una red de área amplia de baja potencia LPWAN es un tipo de red de área amplia de telecomunicaciones inalámbricas diseñada para permitir comunicaciones de largo alcance a una velocidad de bits baja entre objetos conectados.

A diferencia de tecnologías como 2G, 3G que son implementadas para telefonía celular a pesar de tener largo alcance presentan alto consumo para su implementación y otras tecnologías como lo son el Bluetooth y ZigBee son de corto alcance como se muestra en la figura 3.2.

3

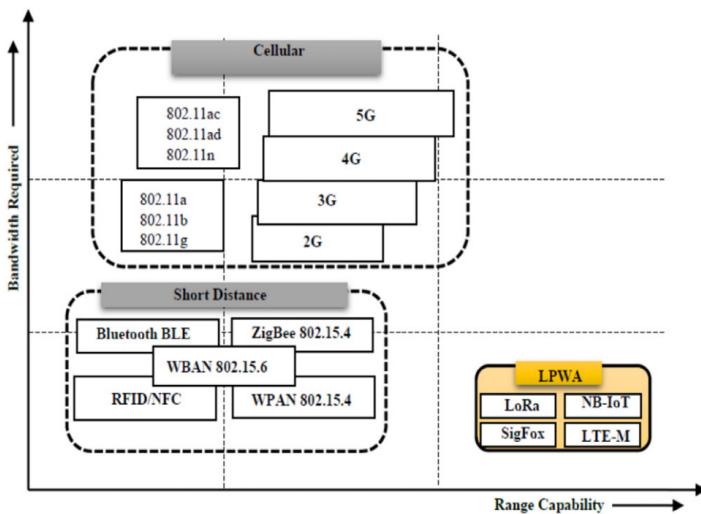


Figura 3.2: Velocidad de trasferencia de datos vs alcance para diferentes tecnologías

Fuente: ([Sinha et al., 2017](#))

Las redes LPWAN son implementadas para IoT debido a los requerimientos de estas entre las cuales se encuentran ser de bajo consumo de energía, largo alcance y baja velocidad para la transmisión de datos, otra de las características de las redes LPWAN es que trabajan en la banda ISM libres sin necesidad de licencia.

Las características principales de una red LPWAN cubrirá una amplia gama de aplicaciones, pero existen requisitos comunes que deben guiar en el diseño de una red LPWAN ([Bardyn et al., 2016](#)).

- La operación de ultra bajo consumo de energía
- Las restricciones económicas son un fuerte impulsor. Los despliegues deben ser baratos para una amplia aceptación; esto significa objetos de bajo costo, sin tarjeta SIM o equivalente, fácil instalación en red y mínimo mantenimiento.
- La complejidad de hardware y software de los objetos debe ser muy limitada, lo que significa arquitecturas simples y protocolos simples.

- El nivel de actividad puede variar de una aplicación a otra, pero para limitar el consumo de energía, la red no debe requerir que el objeto se active, a menos que sea necesario enviar o recibir datos.
- La infraestructura de red debe ser fácil de implementar con una buena cobertura general . Agregar un nuevo objeto a la red deben ser tareas simples. El protocolo debe alinearse con algún tipo de estándar para maximizar la disponibilidad de objetos y una conectividad perfecta.
- La transferencia de datos entre el objeto y el usuario de datos final debe estar completamente protegida. El operador de red no debe poder acceder a datos significativos. Como parte de la seguridad, el enlace de RF también debe ser robusto contra interferencias.
- Los objetos generalmente no se mueven o se mueven lentamente, pero se pueden colocar en entornos con características de canal de rápido movimiento, como estar cerca de una carretera. La modulación debe ser robusta a algún posible desvanecimiento.
- Desde la perspectiva de la aplicación, los objetos proporcionarán datos que se utilizarán para construir una gran variedad de servicios, ya sea directamente o a través de procesos complejos de fusión de datos y aprendizaje automático.

Para aplicaciones LPWAN se tienen diversas tecnologías disponibles en el mercado entre las cuales se encuentran principal mente LoRa,SigFox y NB-IoT:

### 3.2.1. TECNOLOGÍAS LPWAN

#### LORA

LoRaWAN™ define el protocolo de comunicación y la arquitectura del sistema de la red mientras que la capa física LoRa® permite el enlace de comunicación de largo alcance. El protocolo y arquitectura de red tienen la mayor influencia en la determinación de la vida útil de la batería de un nodo, la capacidad de la red, la calidad del servicio, la seguridad, y la variedad de aplicaciones a través de la red. ([Alliance, 2015](#))

LoRa es la capa física utilizada para crear enlaces de comunicaciones de largas distancias. Muchos sistemas usan modulación FSK por se muy eficiente para bajas potencias. LoRa utiliza CSS la cual es una modulación que mantiene las mismas características de baja potencia de FSK pero aumenta el rango distancia de comunicación.

En una LoRaWAN™ los nodos de la red no están asociados con una puerta de enlace específica. En lugar de ello, los datos transmitidos por un nodo es típicamente recibidos por varias puertas de enlace. Cada puerta de enlace reenviará el paquete recibido desde el extremo-nodo para el servidor de red basado en la nube a través de algunos backhaul (ya sea celular,Ethernet, satélite, o Wi-Fi).

Muchas redes desplegadas existentes utilizan una arquitectura de red de malla. En una red de malla, al fin de los nodos individuales transmitirá la información de otros nodos para aumentar el tamaño del alcance de comunicación y de la célula de la red. Si bien esto incrementa el rango, pero también añade complejidad, reduce la capacidad de la red, y la vida útil de la batería de los nodos , la estructura se muestra en la figura. 3.3

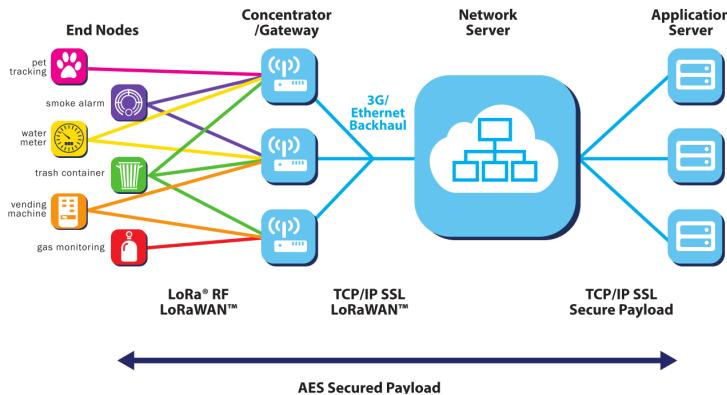


Figura 3.3: Arquitectura de LoRa

Fuente: ([Alliance, 2015](#))

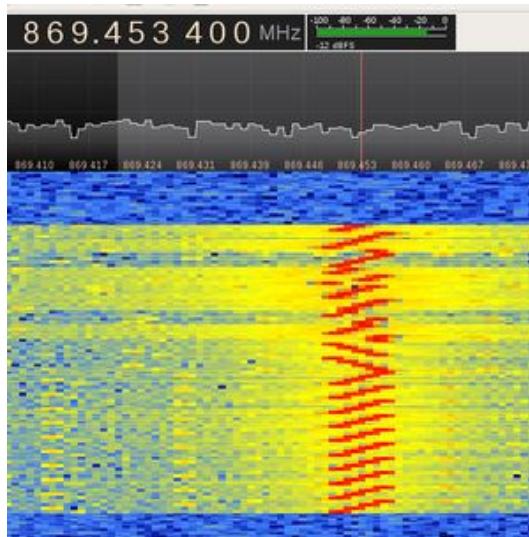
LoRaWAN tiene varias clases diferentes de dispositivos de punto final para abordar las diferentes necesidades que se reflejan en la amplia gama de aplicaciones ([Lora Alliance, 2018](#)):

- Dispositivos finales bidireccionales (Clase A) : Posee dos ventanas para la recepción de datos después de realizar transmisión de datos.
- Dispositivos finales bidireccionales con ranuras de recepción programadas (Clase B): Además de las dos ventanas disponible por la clase A, se pueden programar mas ventanas para recepción de datos.
- Dispositivos finales bidireccionales con ranuras de recepción máxima (Clase C): Tienen ventanas de recepción continuamente abiertas, solo se cierran cuando el dispositivo esta transmitiendo.

#### MODULACION LORA

LoRa es una tecnología de modulación de espectro extendido patentada por Semtech. LoRa es un derivado de Chirp Spread Spectrum (CSS) que reduce la velocidad de datos para mejorar la sensibilidad. Si desea colocar LoRa en la capa OSI, la implementación de la modulación LoRa existe en la capa física y no puede vincularse a otras capas por encima. Esto permite que LoRa se adapte con varias conexiones robustas.

Al igual que el teorema de Shannon-Hartley, un mayor ancho de banda puede compensar la degradación de SNR en el canal de radio. En un sistema Spread Spectrum , los datos enviados se multiplican con Spreading Code , también conocido como C-chip Sequence . La secuencia de chip generalmente tiene una frecuencia mucho más alta que la frecuencia de información que resultó en un ancho de banda más amplio cuando ambas señales se multiplican. En el lado del receptor, la señal recibida se multiplicará con la misma señal de secuencia de chip para obtener los datos originales.



3

Figura 3.4: Espectrograma de señal LoRa mediante lectura SDR  
 Fuente: ([Commons, 2018](#))

La modulación LoRa proporciona alternativas para la comunicación de espectro ensanchado que son de baja potencia y bajo costo en comparación con la técnica convencional de espectro ensanchado. La modulación LoRa utiliza la señal chirp , que es una señal con una frecuencia que varía con el tiempo. Al usar este método, se reducirá la complejidad para el lado del receptor. La señal de salida de esta modulación coincide con el ancho de banda utilizado por la señal chirp.

A parte del ancho de banda, también hay un factor de difusión . Si el Ancho de banda muestra la velocidad del chip, el Factor de dispersión muestra cuánto chip se usa para representar un símbolo. Aparte de eso, el Factor de extensión también muestra cuántos bits contiene un símbolo.

- **Símbolo**

Como se discutió anteriormente, LoRa es una modulación de espectro extendido de chirp . Los datos transmitidos, que es un símbolo, estarán representados por una señal chirp con un rango de frecuencia de  $f_{min}$  a  $f_{max}$  , que se muestra en la Figura 1. En la modulación LoRa, podemos configurar el símbolo cambiando los parámetros de Factor de extensión y Ancho de banda . De acuerdo con la ([Semtech, 2015](#)), un símbolo tomará  $T_s$  de segundo para transmitir, que es una función del ancho de banda(BW) y el factor de dispersión(SF) se puede mostrar con la siguiente ecuación:

$$T_s = \frac{2^{SF}}{BW} \quad (3.1)$$

- **Ancho de banda**

El ancho de banda es el rango de frecuencia de la señal chirp utilizada para transportar los datos de la banda base. En la Figura 1, el ancho de banda se puede ver a partir del ancho de frecuencia utilizado entre  $f_{min}$  a  $f_{max}$ . Aparte de eso, el ancho de banda también puede representar la velocidad de chip de la modulación de señal LoRa.

## 3

$$R_c = BW \quad (3.2)$$

- **Factor de difusión**

El valor del Factor de difusión (SF) determina cuántas fichas se usan para representar un símbolo. Cuanto mayor sea el valor de SF, más chips se usarán para representar un símbolo, lo que significa que habrá más ganancia de procesamiento desde el lado del receptor. Esto permitirá que el receptor acepte señales de datos con un valor negativo de SNR.

$$R_s = \frac{BW}{2^{SF}} \quad (3.3)$$

El factor de dispersión muestra cuántas fichas se utilizan para representar un símbolo, con un factor exponencial de 2. 1 símbolo puede consistir en N chip donde  $N = 2^{SF}$ . Se puede hacer un cambio cíclico para representar un bit y un símbolo enviado. Si hay N cantidad de chips, entonces el valor del símbolo resultante puede variar de 0 a N-1, o ese símbolo 1 puede representar bits SF.

$$R_b = SF * \frac{BW}{2^{SF}} \quad (3.4)$$

- **Tasa de codificación**

La modulación LoRa también agrega una corrección de errores hacia adelante (FEC) en cada transmisión de datos. Esta implementación se realiza codificando datos de 4 bits con redundancias en 5 bits, 6 bits, 7 bits o incluso 8 bits. El uso de esta redundancia permitirá que la señal LoRa soporte interferencias cortas. El valor de la tasa de codificación (CR) debe ajustarse de acuerdo con las condiciones del canal utilizado para la transmisión de datos. Si hay demasiadas interferencias en el canal, se recomienda aumentar el valor de CR. Sin embargo, el aumento en el valor CR también aumentará la duración de la transmisión.

- **Tiempo al aire**

Una consecuencia importante del uso de un factor de dispersión más alto para LoRa es un mayor tiempo en el aire. El módulo LoRa Radio necesita más tiempo para enviar la misma cantidad de datos. Esto significa que el consumo de energía aumenta al aumentar el Factor de difusión.

Base en el espectro extendido LoRa la velocidad de bits nominal de las señales de datos como se muestra en la ecuación 3.5 y la velocidad de chip se puede definir como se muestra en la ecuación 3.6 ([Semtech, 2015](#)).

$$[!ht]R_b = SF \frac{\left[\frac{4}{4+CR}\right]}{\left[\frac{2^{SF}}{BW}\right]} \quad (3.5)$$

$$[!ht]R_b = \frac{BW}{2^{SF}} * 2^{SF} = BW \text{chips/seg} \quad (3.6)$$

3

### SIGFOX

Sigfox es la red de comunicaciones LPWAN (Low-power Wide-area network) específica para IoT más extendida a nivel mundial, con una cobertura próxima al 98% del territorio Europeo y Americano. La red de Sigfox está construida sobre una modulación ultra narrow band (UNB) y opera en la banda de 868MHz en Europa y en la banda de 902MHz en Estados Unidos.

Esta banda de frecuencias permite sólo los mensajes de menor tamaño: 12 bytes por mensaje enviado. La red SIGFOX es bidireccional lo que significa que los objetos conectados a la red pueden enviar y recibir mensajes. Sumado a esto, la red permite sólo un mensaje por objeto cada 10 minutos máximo (140 mensajes por día). Por último, la red SigFox se limita a una velocidad de 100 bit / s. Cada objeto tiene un identificador único en la red. Todos los mensajes transmitidos están firmados con este ID (así como la de la estación que recibe el mensaje) ([SigFox, 2019](#)).

Una de las ventajas de SigFox es su despliegue y cobertura es casi global con una gran facilidad para manejar datos en la nube, y posee una gran soporte por parte de la plataforma Azure de Microsoft que permite acelera en gran medida la ejecución de un proyectos.

El bajo coste de esta tecnología, su aceptación por los fabricantes de dispositivos, o el que sea una red bidireccional son otros factores a favor. Por el contrario, al ser una frecuencia no licenciada podría encontrarse en un futuro fuera de mercado, ya que esta frecuencia podría ser regulada por los organismos públicos y adquirida por el sector de las grandes empresas de telecomunicaciones, las cuales quieran apostar por M2M o NB IoT. ([EFOR, 2018](#))

### NB-IoT

La NarrowBand IoT (NB IoT) es otra red con tecnología LPWAN, en este caso, la gran apuesta de las operadoras de telecomunicaciones a nivel global. Esta tecnología tiene su factor diferencial en que su espectro de funcionamiento entra dentro del rango del LTE o 4G, por lo que su despliegue y explotación comercial está casi asegurada gracias a la red actualmente desplegada. No obstante, el despliegue de la red, la puesta en marcha de esta tecnología y las bondades de la misma están pendientes de ser analizadas por los expertos y por los propios clientes, ya que, por ejemplo, esta red en España está desplegada sólo en Madrid y Valencia a modo de test, aunque se espera que a lo largo de 2017 nuevos nodos pongan en funcionamiento la red de NB IoT. ([EFOR, 2018](#))

B-IoT ocupa un ancho de banda de frecuencia de 200 KHz, que corresponde a un bloque de recursos en la transmisión GSM y LTE . Con esta selección de banda de frecuencia, son posibles los siguientes modos de operación:

- Operación independiente: un posible escenario es la utilización de las bandas de frecuencias GSM utilizadas actualmente.
- Operación en banda de protección: utilizando los bloques de recursos no utilizados dentro de la banda de guarda de un operador LTE.
- Operación en banda: utilizando bloques de recursos dentro de un operador LTE

### 3.2.2. COMPARACIÓN EN TÉRMINOS DE FACTORES IOT

Una LAN inalámbrica debe cumplir los mismos requisitos típicos de cualquier otra red LAN, incluyendo alta capacidad, cobertura de pequeñas distancias, conectividad total de las estaciones conectadas y capacidad de difusión. Además, existe un conjunto de necesidades específicas para entornos de LAN inalámbricas. Entre las más importantes se encuentran las siguientes:

- **Rendimiento:** el protocolo de control de acceso al medio debería hacer un uso tan eficiente como fuera posible del medio no guiado para maximizar la capacidad.
- **Número de nodos:** las LAN inalámbricas pueden necesitar dar soporte a cientos de nodos mediante el uso de varias celdas.
- **Conexión a la LAN troncal:** en la mayoría de los casos es necesaria la interconexión con estaciones situadas en una LAN troncal cableada. En el caso de LAN inalámbricas con infraestructura, esto se consigue fácilmente a través del uso de módulos de control que conectan con ambos tipos de LAN. Puede ser también necesario dar soporte a usuarios móviles y redes inalámbricas.
- **Área de servicio:** una superficie de cobertura para una red LAN inalámbrica tiene un diámetro típico de entre 100 y 300 metros.
- **Consumo de batería:** los usuarios móviles utilizan estaciones de trabajo con batería que necesitan tener una larga vida cuando se usan con adaptadores sin cable. Esto sugiere que resulta inapropiado un protocolo Media Access Control (MAC) que necesita nodos móviles para supervisar constantemente los puntos de acceso o realizar comunicaciones frecuentes con una estación base.
- **Robustez en la transmisión y seguridad:** a menos que exista un diseño apropiado, una LAN inalámbrica puede ser propensa a sufrir interferencias y escuchas. El diseño de una LAN inalámbrica debe permitir transmisiones fiables incluso en entornos ruidosos y debe ofrecer cierto nivel de seguridad contra escuchas.
- **Funcionamiento de red ordenada:** medida que las LAN inalámbricas se están haciendo más populares, es probable que dos o más de estas redes operen en la misma o en alguna zona en que sea posible la interferencia entre ellas. Estas interferencias pueden frustrar el normal funcionamiento del algoritmo MAC y pueden permitir accesos no autorizados a una LAN particular.
- **Funcionamiento sin licencia:** los usuarios podrían preferir adquirir y trabajar sobre LAN inalámbricas que no precisan de una licencia para la banda de frecuencia usada por la red.
- **Sin intervención/nómada:** el protocolo MAC usado en LAN inalámbricas debería

permitir a las estaciones móviles desplazarse de una celda a otra.

- **Configuración dinámica:** los aspectos de direccionamiento MAC y de gestión de red de la LAN deberían permitir la inserción, eliminación y traslado dinámicos y automáticos de sistemas finales sin afectar a otros usuarios.

Se deben considerar muchos factores al momento de elegir cualquiera a la tecnología LPWAN para una aplicación de IoT que incluye calidad de servicio, duración de la batería, latencia, escalabilidad, longitud de la carga útil, cobertura, rango, implementación y costo. En la tabla 3.1 se comparan Sigfox, LoRa y NB-IoT en términos de estos factores y sus diferencias técnicas.

Factor IoT	SigFox	LoRaWan	NB-IoT
Modulación	BPSK	FSK,CSS	QPSK
Frecuencia	Sin Licencia 868 MHz Europa 915 MHz América 433 MHz Asia	Sin Licencia 868 MHz Europa 915 MHz América 433 MHz Asia	Licenciada bandas LTE
Ancho de Banda	100 Hz	250KHz 125 KHz	250 KHz
Max Tasa Bits	100 bps	50 Kbps	200 Kbps
Bidireccional	Limitada Half-Duplex	Si Half-Duplex	Half-Duplex
Max Mensajes/Día	140(UL) 4(DL)	Sin Limite	Sin Limite
Max Longitud de Trama	12 bytes(UL) 8 bytes(DL)	243 bytes	1600 bytes
Rango	10 Km (Urbano) 40 Km (Rural)	5 Km (Urbano) 20 Km (Rural)	1 Km (Urbano) 10 Km (Rural)
Inmunidad a Interferencia	Muy Alta	Muy Alta	Baja
Autenticación y Encriptación	No Soportada	Si AES 128b	Si LTE
Tasa Bits Variable	No	Si	No
HandOver	Los dispositivos no se unen a una solo base	Los dispositivos no se unen a una solo base	Los dispositivos se unen a una solo base
Ubicación	Si(RSSI)	Si(RDOA)	No
Permite Red Privada	No	Si	No
Estándar	Sigfox, ETIS	LoRa-Alliance	3GPP

Cuadro 3.1: Visión general de las tecnologías LPWAN: Sigfox, LoRa y NB-IoT

Fuente: ([Mekki et al., 2019](#))

### 3.2.3. CONTROL DE ACCESO AL MEDIO

Uno de los problemas a resolver en un sistema de comunicaciones es cómo repartir entre varios usuarios el uso de un único canal de comunicación o medio de transmisión, para que puedan gestionarse varias comunicaciones al mismo tiempo. Sin un método de organización, aparecerían interferencias que podrían resultar molestas o directamente impedir la comunicación. Este concepto se denomina multiplexado o control de acceso al medio, según el trayecto de comunicación.

Más específicamente, en redes de computadoras, la sigla MAC , del inglés Media Access Control, se emplea en la familia de estándares IEEE 802, para definir la subcapa de control de acceso al medio.

La subcapa MAC se sitúa en la parte inferior de la capa de enlace de datos (capa 2 del Modelo de Referencia OSI). La implementación exacta de esta subcapa puede variar dependiendo de los requerimientos de la capa física (por ejemplo Ethernet, Token Ring, WLAN).

Algunas de las funciones de la subcapa MAC incluyen:

- Controlar el acceso al medio físico de transmisión por parte de los dispositivos que comparten el mismo canal de comunicación.
- Agregar la dirección MAC del nodo fuente y del nodo destino en cada una de las tramas que se transmiten.
- Al transmitir en origen debe delimitar las tramas de red agregando bits de bandera (flags) para que el receptor pueda reconocer el inicio y fin de cada trama.
- Al recibir en destino debe determinar el inicio y el final de una trama de datos dentro de una cadena de bits recibidos por la capa física.
- Efectuar detección y, si procede, corrección de errores de transmisión.
- Descartar tramas duplicadas o erróneas.

3

Existen dos tipos básicos de métodos de control de acceso al medio en medios para redes inalámbricas:

- Distribuido (tipo CSMA/CD): Nodos deciden cuando transmiten mediante sistema de detección de portadora. Adecuado en redes ad-hoc con estaciones paritarias y redes con tráfico a ráfagas.
- Centralizado: Existe gestión centralizada de transmisión. En redes inalámbricas de infraestructura (conectada a LAN cableada). Adecuado cuando existen datos sensibles al tiempo o de alta prioridad.

Existen dos tipos básicos de métodos de control de acceso al medio en medios compartidos:

- Controlado: Cada nodo tiene su propio tiempo para utilizar el medio. Los dispositivos de red toman turnos para acceder al medio. Cuando un dispositivo coloca una trama en los medios, ningún otro dispositivo puede hacerlo hasta que la trama haya llegado al destino y haya sido procesada por el destino.
- Basado en la contención:

Todos los nodos compiten por el uso del medio. Permite que cualquier dispositivo intente acceder al medio siempre que haya datos para enviar. El uso de este nombre es debido a que los equipos de la red se contienen hasta que haya una oportunidad para enviar los datos. A este tipo de métodos de acceso al medio se les conoce también como métodos aleatorios.

#### TIPOS DE MÉTODOS DE ACCESO CONTROLADOS

Las dos variantes principales son el método de sondeo y el

- Método de sondeo (polling) Se designa un nodo como maestro y se encargará de dirigir los turnos. Para que un nodo pueda transmitir debe recibir permiso del nodo central a través de un mensaje de sondeo. Este permiso va pasando continuamente de estación en estación. Cada estación puede transmitir cuando recibe el

permiso y encuentra el medio libre. Al finalizar su transmisión el nodo maestro pasa el permiso a la estación siguiente y así sucesivamente.

- **Método de paso de testigo (token passing)** En esta técnica no hay ningún nodo maestro, pero si una trama especial de pequeño tamaño llamada testigo (token) que va siendo intercambiada entre los nodos según un orden preestablecido. Un nodo puede transmitir cuando tiene la trama testigo y, mientras no la tenga, deberá esperar. Este método de acceso al medio a sido ampliamente utilizado en las redes con topología en anillo.

3

#### TIPOS DE MÉTODOS DE ACCESO AL MEDIO BASADOS EN CONTENCIÓN

- **CSMA/CD**

En CSMA/CD, el dispositivo monitorea los medios para detectar la presencia de una señal de datos:

Si hay una señal el dispositivo espera hasta que encuentre el canal libre.

Si no hay una señal de datos, esto indica que el medio está libre, el dispositivo transmite los datos. Durante la emisión se sondea el medio para detectar si se produce una colisión. Si se produce una colisión, todos los dispositivos dejan de enviar y lo intentan después de un tiempo de espera aleatorio (423 ms en la imagen de ejemplo).

Las formas tradicionales de Ethernet usan este método.

- **CSMA/CA**

En CSMA/CA, el dispositivo examina los medios para detectar la presencia de una señal de datos:

Si el medio está libre: el dispositivo envía una notificación a través del medio, sobre su intención de utilizarlo. El dispositivo luego envía los datos.

Este método es utilizado por las tecnologías de redes inalámbricas 802.11. y lo veremos con más detalle cuando estudiemos las redes inalámbricas

### 3.3. COMPUTADORAS DE PLACA REDUCIDA

Una placa computadora u ordenador de placa reducida SBC es una computadora completa en un sólo circuito. El diseño se centra en un sólo microprocesador con la RAM, E/S y todas las demás características de un computador funcional en una sola tarjeta que suele ser de tamaño reducido, y que tiene todo lo que necesita en la placa base.

La principal característica de las SBC son sus reducidas dimensiones. Van desde tamaños similares a una memoria USB, hasta medidas parecidas a una tarjeta de visita o de crédito, como la Raspberry Pi que mide 8,5 x 5,3 cm. Otra de las características de las placas SBC es su precio. Por lo general son muy económicas, en comparación con otras soluciones que se pueda encontrar.

Estas placas ofrecen poca potencia, aunque esto es relativo. Actualmente todas las placas SBC ofrecen una potencia más que suficiente para ofimática, desarrollo e incluso reproducción multimedia. Si bien es posible correr emuladores y juegos livianos, aún no cuentan con la potencia y características suficientes tener un uso puro para juegos actuales de demasiados requerimientos.

Debido a las grandes niveles de integración y reducción de componentes y conectores, los computadores en una tarjeta suelen ser más pequeños, livianos, más confiables y con un mejor manejo de la potencia eléctrica que los computadores de múltiples tarjetas.

Por otro lado, esto implica que actualizar uno de estos sistemas es normalmente imposible. Si hay un fallo o se necesita una actualización, es normal que toque reemplazar la tarjeta completa.

Entre las principales SBC se encuentran:

- **Raspberry Pi**: La placa SBC más popular se llama Raspberry Pi. Cuenta con varias versiones y que tiene una gran comunidad detrás. El proyecto surgió en una universidad de Inglaterra, para buscar un hardware económico y libre para enseñar computación en escuelas. Actualmente y gracias a su activa comunidad, se puede hacer casi cualquier cosa con esta placa, desde un servidor hasta la posibilidad de ser el hardware que mueve a una tablet.
- **Arduino** : SBC de bajo coste y desarrollo libre (licencia Creative Commons), de origen italiano. Se utiliza en una gran diversidad de proyectos, incluidos para CPU de impresoras 3D, o para control de sistemas de riego automatizado por ejemplo.
- **BeagleBone Black** : Es la alternativa estadounidense a Raspberry Pi. No existe, en general, mucha diferencia entre la potencia de esta placa con el resto, pero ésta, puede soportar Ubuntu o funcionar como un accesorio más del PC tradicional.
- **PcDuino** : Está basado en los esquemas de Arduino, pero a diferencia del resto, es bastante grande, llega a los 12 cm de largo por 6 cm de ancho. Esta placa admite y soporta Ubuntu y Android.
- **Pandaboard** : Es quizás la menos famosa, pero no por ello la menos interesante. Tiene una gran comunidad que está creando interesantes proyectos con esta placa SBC. Permite la conexión wifi gracias a una antena incorporada a la placa. Característica que no todas estas placas tienen

Model	Raspberry Pi3 modelo B+	Raspberry Pi Zero Wireless	Raspberry Pi Zero v1.3
RAM	1 GB	512 MB	512 MB
USB Port	4	1 micro USB	1 micro USB
GPIO	17 x GPIO UART SPI I2C 1-Wire EEPROM 3.3V/5V/GND	StartFragment 17 x GPIO UART SPI I2C 1-Wire EEPROM 3.3V/5V/GND EndFragment	StartFragment 17 x GPIO UART SPI I2C 1-Wire EEPROM 3.3V/5V/GND EndFragment
Memory	Mirco SD	Mirco SD	Mirco SD
WI-FI	No	Si	No
Ethernet	Si	No	NO

Cuadro 3.2: Comparación de placas Raspberry

Fuente: ([ModMyPi LTD, 2019](#))

### 3.4. TOPOLOGÍAS DE RED

Hay muchas maneras de organizar los componentes de telecomunicaciones para formar una red, y por lo tanto, hay múltiples clasificaciones de redes. Una manera de describir las redes es por su forma o topología, es decir, el diseño físico de nodos en una red.

La palabra topología es un término de origen griego que se refiere al estudio de las formas y que se emplea en el diseño de redes de comunicación, para referirse precisamente a la forma en que están conectados los nodos de una red. Precisamente una red de comunicaciones esta formada básicamente por tres elementos: Nodo, Enlaces y Equipos terminales. Por lo tanto, definiremos como topología de una red a la forma como los equipo terminales se conectan entre sí y con los nodos, a través de los enlaces de comunicaciones.

Normalmente, las redes remotas como locales, se apoyan en la topología siguiente: estrella, malla, anillo, bus y árbol. A estas se les denomina topologías básicas pero existen por combinación de ellas las topologías mixtas o combinadas.

Las redes en general, consisten en “compartir recursos”, y uno de sus objetivos es hacer que todos los programas, datos y equipos estén disponibles para cualquiera de la red que así lo solicite, sin importar la localización física del recurso y del usuario. Un segundo objetivo consiste en proporcionar una alta fiabilidad al contar con fuentes alternativas de suministro. Un punto muy relacionado es la capacidad para aumentar el rendimiento del sistema en forma gradual a medida que crece la carga, simplemente añadiendo mas procesadores. Además la red puede proporcionarnos un poderoso medio de comunicación entre personas que se encuentran muy alejadas una de otras ([Aakvaag and Frey, 2006](#)).

Una red ad hoc inalámbrica es un tipo de red inalámbrica descentralizada. La red es ad hoc porque no depende de una infraestructura preexistente, como routers (en redes cableadas) o de puntos de accesos en redes inalámbricas administradas. En lugar de ello, cada nodo participa en el encaminamiento mediante el reenvío de datos hacia otros nodos, de modo que la determinación de estos nodos hacia la información se hace dinámicamente sobre la base de conectividad de la red. Además del encaminamiento clásico, las redes ad hoc pueden usar un flooding (inundación de red) para el reenvío de datos.

Las redes ad-hoc se refiere típicamente a cualquier conjunto de redes donde todos los nodos tienen el mismo estado dentro de la red y son libres de asociarse con cualquier otro dispositivo de red ad hoc en el rango de enlace. Las redes ad hoc se refieren generalmente a un modo de operación de las redes inalámbricas IEEE 802.11.

También se refiere a la habilidad de un dispositivo de red de mantener la información del estado de conexión para cualquier cantidad de dispositivos en un rango de un enlace (o "salto.<sup>en</sup> argot de informática), y por lo tanto, es más a menudo una actividad de capa 2. Debido a esta única actividad de capa 2, las redes ad hoc por sí solas no soportan un ambiente de red con IP encaminable sin las capacidades adicionales de otra capa 2 o capa 3.

Este tipo de red permite la adhesión de nuevos dispositivos y así, con el solo hecho de estar en el rango de alcance de un nodo ya perteneciente a la red establecida. El protocolo que rige este tipo de comunicaciones es el 802.11, que define todos los parámetros necesarios para establecer la comunicación entre dispositivo inalámbricos. El principal inconveniente de este tipo de redes radica en el número de saltos que debe recorrer la información antes de llegar a su destino. Cada nodo que retransmite la información implica un salto, cuanto más saltos mayor es el tiempo que tarda en llegar la información a su destino y aumenta la probabilidad de que la información se corrompa con cada salto.

3

Diseño de topología en redes Hay una topología adecuada para cada fin, al escoger la topología que implantaremos hay que tener en cuenta las ventajas y desventajas que tiene cada una de ellas.

### 3.4.1. ESTRELLA

En esta topología todos los nodos son conectados a un dispositivo central llamado hub. Los datos de una computadora son transmitidos por el hub al resto de las computadoras en red. La ventaja de esta topología es que todos los procesos son centralizados y esto permite un fácil control de tráfico un diseño se ilustra en la figura 3.5.

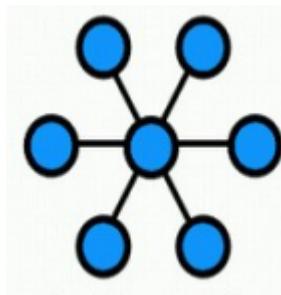


Figura 3.5: Red Configurada en estrella  
Fuente ([wikipedia, 2019](#))

#### Ventajas

- Estructura simple
- Cada Nodo es independiente de los demás
- Facilidad para detectar Nodos que estén causando problema en la red
- Fácil conexión a la red
- Son las mejores para aplicaciones que estén ligadas a gran capacidad de procesamiento
- Permite añadir nuevas computadoras a la red.
- Control de tráfico centralizado
- La falta de un nodo no afecta a la red.

#### Desventajas

- Limitación en rendimiento y confiabilidad.
- Su funcionamiento depende del servidor central.
- Su crecimiento depende de la capacidad del servidor central.
- La distancia entre las estaciones de trabajo y el servidor.

3

### 3.4.2. MALLA

La topología en malla principalmente nos ofrece redundancia. En esta topología todas los nodos están interconectadas entre sí. Esta configuración provee redundancia porque si un nodo falla hay otros que permiten mantener la comunicación. Muchas veces la topología MALLA se va a unir a otra topología para formar una topología híbrida. Las redes en malla son aquellas en las cuales todos los nodos están conectados de forma que no existe una preeminencia de un nodo sobre otros, en cuanto a la concentración del tráfico de comunicaciones. En muchos casos la malla es complementada por enlaces entre nodos no adyacentes, que se instalan para mejorar las características del tráfico. Este tipo de redes puede organizarse con equipos terminales solamente (en lugar de nodos), para aquellos casos en que se trate de redes de transmisión de datos. Estas redes permiten en caso de una iteración entre dos nodos o equipos terminales de red, mantener el enlace usando otro camino con lo cual aumenta significativamente la disponibilidad de los enlaces, en la figura 3.6 al lado izquierdo se muestra una red en malla simple y al lado derecho se muestra una red en malla totalmente conectada .

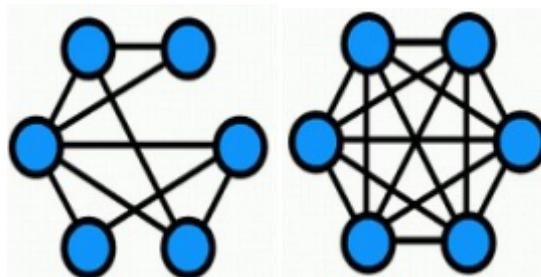


Figura 3.6: Red Configurada en Malla  
Fuente ([wikipedia, 2019](#))

- Baja eficiencia de las conexiones o enlaces, debido a la existencia de enlaces redundantes.
- Por tener redundancia de enlaces presenta la ventaja de posibilitar caminos alternativos para la transmisión de datos y en consecuencia aumenta la confiabilidad de la red.
- Como cada estación esta unida a todas las demás existe independencia respecto de la anterior.
- Poco Control y realización demasiado complejo pero maneja un grado de confiabilidad aceptable.

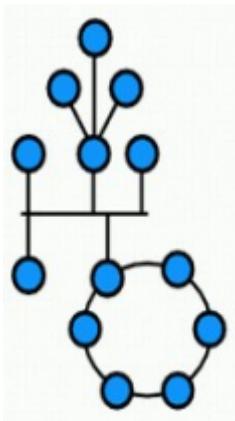


Figura 3.7: Red Hibrida  
Fuente ([wikipedia, 2019](#))

### 3.4.3. HÍBRIDAS

La topología híbrida es una de las más frecuentes y se deriva de la unión de varios tipos de topologías de red, de aquí el nombre de híbridas.. Ejemplos de topologías híbridas serían: en árbol, estrella-estrella, bus-estrella, etc. En una topología híbrida, se combinan dos o más topologías para formar un diseño de red completo. Raras veces, se diseñan las redes utilizando un solo tipo de topología en la figura 3.7. Por ejemplo, es posible que desee combinar una topología en estrella con una topología de bus para beneficiarse de las ventajas de ambas.

Ventajas:

- Si falla algún nodo no le pasa nada a la red
- Rapidez de acceso a la red

Desventajas:

- Si el concentrador principal se cae, se viene abajo toda la red
- Su costo de implementación es muy elevado



# 4

## METODOLOGÍA

Se pretende alcanzar los objetivos de este trabajo con la aplicación de la información encontrada en los diversos artículos citados en el presente documento. El prototipo que se pretende implementar requiere los siguientes pasos en los cuales se analizara y desarrollara las diferentes partes que componen al dispositivo.

### 4.1. ANÁLISIS DE REQUERIMIENTOS

El primer objetivo consiste en evaluar y seleccionar diferentes tipos de receptores de posicionamiento global como de módulos que permitan implementar redes LPWAN. Para cumplir este objetivo se analizarán las características de los receptores GNSS de un fabricante en concreto, U-blox. Esta decisión se debe a que cada fabricante de este tipo de receptores posee modelos para todo tipo de aplicaciones, cada uno con diferentes especificaciones. Por esta razón se consideró simplemente evaluar las características de los diferentes modelos de un solo fabricante para reducir el espectro de búsqueda y tiempo a la hora de seleccionar el receptor a usar, los receptores indicados para el tipo de trabajo a realizar son los receptores U-blox de la serie M8T y NEO-6T, debido a que estos son compatibles con la librería con la cual se pretende trabajar. El proceso de selección del fabricante se realizó simplemente observando la presencia de este en el mercado y su variedad de receptores.

La selección y evaluación de los módulos para la red LPWAN se hará pensando primordial mente en la red de sensores como tal. Las redes actualmente conocidas tienen características únicas para inclinarse por una u otra, cada una ofreciendo diferentes alcances y anchos de banda entre otras características, como cada una posee sus especificaciones no permiten el uso de cualquier dispositivo de radiofrecuencia, cada uno requiere diferentes módulos, para la implementación de la red.

## 4.2. DISEÑO E IMPLEMENTACIÓN DE LA RED

Se analizara las deferentes topologías de red en base a sus ventajas y desventajas así como los diferentes métodos de acceso al control al medio, buscando la que mejor se adapte al problema presentado, y que el dispositivo seleccionado anteriormente permita implementar, se diseñaran los módulos que componen la red, como son las diversas conexiones para su correcto funcionamiento tanto para el envío de datos como para la lectura de información desde el modulo GNSS. Este objetivo se logrará con el desarrollo de los módulos y la selección del tipo de conexión para la red.

## 4.3. DESARROLLO DE INTERFAZ GRÁFICA

Implementar un software que realice el proceso de comunicación interna de cada componente de los nodos de la red para su posterior visualización y trasferencia de datos hacia los otros puntos de la red. Este objetivo se logrará con el desarrollo de una interfaz con la cual se pretenderá controlar y observar los datos que hay en la red. La implementación de este software para la Raspberry Pi se encargara de generar un entorno gráfico y todos los cálculos concernientes para la administración de la red y sus datos. Este software de esta manera terminara por consolidar el prototipo al poder visualizar los datos y ser accesibles en cualquier punto de la red.

## 4.4. PRUEBAS DE CAMPO

Realizar las pruebas de campo que permitan verificar la funcionalidad de la red de sensores.

Las pruebas se iniciarán en tres fases.

- La primera fase consiste en implementar la red con los módulos LPWAN, utilizando una SBC para verificar su funcionalidad.
- La segunda fase consiste en iniciar las Pruebas de posicionamiento tras la adquisición de los receptores.
- La tercera fase se llevará a cabo con la integración de las dos fases anteriores funcionando con Raspberry PI y en campo abierto.

Para realizar la verificación del funcionamiento de los nodos de la red se tomo se realizaron varias pruebas independientes donde se realizaron pruebas de tiempos para la trasmisión de datos y la confiabilidad de la red para trasmitir datos, como tambien se realizo una comparación con arquitecturas de red que presentan similitudes con la red planteada.

# 5

## DESARROLLO

En base a la metodología planteada se desarrollaron actividades para el cumplimiento de los objetivos separados en partes ,en las cuales primero se realizo el análisis de requerimientos para la implementación de los módulos teniendo en cuenta el modulo de comunicaciones , la placa de desarrollo para la implementación y el modulo gnss, posteriormente se realizo el el análisis de la arquitectura de red y el desarrollo para la implementación así como el desarrollo de la interfaz gráfica para la visualización de la información .

### 5.1. ANÁLISIS Y REQUERIMIENTOS

En esta sección se realizo un análisis de los requerimientos sobre los dispositivos GNSS , LPWAN y SBC que cumplan con los requisitos que se plantean para el desarrollo del proyecto y se ilustro el dispositivo seleccionado y sus características.

#### 5.1.1. DISPOSITIVO PARA EL POSICIONAMIENTO

En base a las características de los GNSS vistas en el marco teórico se plantearon los siguientes requerimientos para la selección del dispositivo:

##### REQUERIMIENTOS

- Conexión con diversas constelaciones GPS y GLONASS.
- Protocolos de comunicación I2C,SPI o UART.
- Capacidad de entregar datos en un formato RAW(low-level data) basado en el protocolo(NMEA).
- Alta frecuencia para la actualización de datos.
- Alta precisión y exactitud.
- Antena Incorporada.
- Compatible con RTKLIB.

## DISPOSITIVO SELECCIONADO

Con base en estas características deseadas se realizo una búsqueda sobre diferentes dispositivos GNSS que cumplieran estas condiciones , principalmente de la marca U-Blox ya que la mayoría de sus productos tienen la capacidad de entregar datos en formato RAW(low-level data) .

Tras realizar la búsqueda respectiva sobre los diversos dispositivos se selecciono el modulo GNSS "ZED-F9P"la cual esta disponible con la placa de desarrollo C099-F9P Application Board"la cual posee diversos protocolos de comunicación, el modelo fisico se ilustra en la figura 5.1 y otras características en el cuadro 5.1.

**5**

Parameter	Specification							
Receiver type	Multi-band GNSS high precision receiver							
Accuracy of time pulse signal	RMS 99%	30 ns 60 ns						
Frequency of time pulse signal		0.25 Hz to 10 MHz (configurable)						
Operational limits <sup>1</sup>		Dynamics	$\leq 4 \text{ g}$					
		Altitude	50,000 m					
		Velocity	500 m/s					
Velocity accuracy <sup>2</sup>			0.05 m/s					
Dynamic heading accuracy <sup>2</sup>			0.3 deg					
<b>GNSS</b>		<b>GPS+GLO+GAL +BDS</b>	<b>GPS+GLO+GAL</b>	<b>GPS+GAL</b>	<b>GPS+GLO</b>	<b>GPS+BDS</b>	<b>GPS</b>	
Acquisition <sup>3</sup>	Cold start	24 s	25 s	29 s	26 s	28 s	29 s	
	Hot start	2 s	2 s	2 s	2 s	2 s	2 s	
	Aided start <sup>4</sup>	2 s	2 s	2 s	2 s	2 s	2 s	
Nav. update rate	RTK	8 Hz	10 Hz	15 Hz	15 Hz	15 Hz	20 Hz	
	PVT	10 Hz	12 Hz	20 Hz	25 Hz	25 Hz	25 Hz	
	RAW	20 Hz	20 Hz	25 Hz	25 Hz	25 Hz	25 Hz	
Convergence time <sup>5</sup>	RTK	< 10 s	< 10 s	< 10 s	< 10 s	< 10 s	< 30 s	

Cuadro 5.1: ZED-F9P Rendimiento en diferentes modos GNSS

Fuente: ([u blox, 2019](#))



Figura 5.1: C099-F9P Application Board

Fuente: ([u blox, 2019](#))

### 5.1.2. DISPOSITIVO PARA EL PROCESAMIENTO

En base a las características de las SBC vistas en el marco teórico se plantearon los siguientes requerimientos para la selección del dispositivo:

#### REQUERIMIENTOS

Para la implementación de los módulos se opto por la SBC Raspberry Pi ya que esta placa es la de mayor facilidad para el desarrollo de las diversas conexiones que se requieren entre las cuales se encuentran comunicación SPI, UART, Serial como la posibilidad de crear de manera local una red para la distribución de la información(AP wi-fi) .

En base a la comparación establecida en la tabla 3.2 se observa que una de las SBC que cumple con los requisitos sin necesitar componentes son la Raspberry Pi Zero Wireless , Raspberry Pi 3 Model A+ y Raspberry Pi 3 Model B+, en base a que para esta red no se requiere una alto poder de procesamiento de datos y se busca que el consumo sea el menor posible se opto por la Raspberry Pi Zero Wireless

#### DISPOSITIVO SELECCIONADO

El Raspberry Pi Zero W extiende la familia Pi Zero. Lanzado a finales de febrero de 2017, el Pi Zero W tiene toda la funcionalidad del Pi Zero original, pero viene con conectividad adicional, que consiste en ([Raspberry, 2018](#)):

- 802.11 b/g/n wireless LAN
- Bluetooth 4.1
- Bluetooth Low Energy (BLE)
- 1GHz, single-core CPU
- 512MB RAM
- Mini HDMI and USB On-The-Go ports
- Micro USB power
- HAT-compatible 40-pin header
- Composite video and reset headers
- CSI camera connector

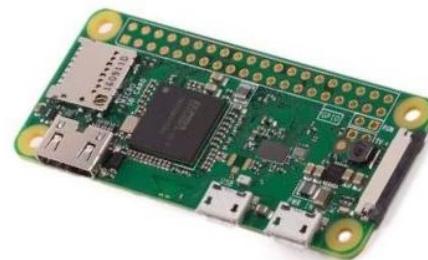


Figura 5.2: Raspberry Pi Zero Wireless

Fuente: ([Raspberry, 2018](#))

## 5.2. DISPOSITIVO DE COMUNICACIONES

Para la realizar la selección del dispositivo LPWAN a utilizar en la red se tuvieron en cuenta varias características propias del dispositivo y redes inalámbricas para lo cual se tomaron los siguientes indicadores:

### 5.2.1. CALIDAD DEL SERVICIO

Sigfox y LoRa emplean espectros sin licencia y Protocolos de comunicación asíncronos, alta inmunidad a interferencias. Sin embargo, no pueden ofrecer la misma calidad de servicio proporcionada por NB-IoT. NB-IoT emplea un espectro con licencia y un protocolo síncrono basado en LTE, que son óptimos para calidad de servicio(QoS) a expensas del costo, NB-IoT se prefiere para aplicaciones que requieren una calidad de servicio garantizada, mientras que las aplicaciones que no tienen esta restricción deben elegir LoRa o Sigfox.

Para el presente caso ya que NB-IoT trabaja con bandas licenciadas el se presentaría un costo para elevado para la implementación de esta por lo cual se optara principalmente por LoRa o SigFox.

5

### 5.2.2. LATENCIA

Una de las factores para seleccionar el modelo LPWAN a implementar son los tiempos de retardo para el envío y recepción de la información, NB-IoT ofrece la ventaja de una baja latencia,A diferencia de Sigfox, LoRa tiene la posibilidad de tener varios modos de comunicación entre los cuales están la clase A y C ,la clase C se utiliza para manejar una latencia bidireccional baja a expensas del aumento del consumo de energía. Por lo tanto, para las aplicaciones que son insensibles a la latencia y que no tienen una gran cantidad de datos para enviar, Sigfox y LoRa de clase A son las mejores opciones. Para aplicaciones que requieren baja latencia NB-IoT y LoRa clase C son las mejores opciones.

### 5.2.3. COBERTURA DE RED Y ALCANCE

La mayor ventaja de utilización de Sigfox es que toda una la ciudad puede ser cubierta por una sola estación base (<40 km rural, <10Km urbano). Por el contrario, LoRa tiene un rango inferior (<20 km rural , <5 km urbano). NB-IoT tiene el rango más bajo y las capacidades de cobertura (<10 km rural,<1km urbano). Además, el despliegue de NB-IoT se limita a las estaciones base LTE. Por lo tanto, no es adecuado para regiones rurales o suburbanas que no se benefician de la cobertura de LTE.

Para la red a implementar se enfocara principalmente a regiones rurales o suburbanas para lo cual LoRa o SigFox no presenta problemas en la implementación.

### 5.2.4. MODELO DE DESPLIEGUE Y DISPONIBILIDAD

Las especificaciones NB-IoT se publicaron en junio de 2016; por lo tanto, se necesitará tiempo adicional antes de que se establezca su red. Sin embargo, los ecosistemas

de Sigfox y LoRa están maduros y ahora están bajo comercialización en varios países y ciudades. LoRa tiene la ventaja que le permite implementarse actualmente en 51 países en comparación con 60 países para Sigfox ([Lora-Alliance, 2019](#)),([SigFox, 2019](#)). Sin embargo, las implementaciones mundiales de LoRa y Sigfox todavía están en desarrollo.

Además, una ventaja significativa del ecosistema LoRa. Es su flexibilidad. A diferencia de Sigfox y NB-IoT, LoRa ofrece una implementación de red local, es decir, una LAN que utiliza la puerta de enlace LoRa, así como la operación de la red pública a través de estaciones base.

LoRa al permitir despliegues locales, para el caso de la red a implementar esto sería lo mejor ya que no en todo caso se tendrán acceso a puntos de Ethernet para el procesamiento de la nube de los datos, estos se procesaran de manera local.

### 5.2.5. ESCALABILIDAD Y LONGITUD DE CARGA ÚTIL

Una de las características mas importantes de una red LPWAN es la escalabilidad de la red y la capacidad de manejar un gran numero de dispositivos NB-IoT al estar basada en LTE tiene una mayor facilidad al momento de su expansión con la red ya implementada (LTE/4G), NB-IoT permite una conectividad hasta 100K dispositivos finales por celda en comparación con 50k por celda para SigFox y LoRa.

NB-IoT también ofrece la ventaja de máximo longitud de carga útil. Como se presenta en la tabla 3.1, NB-IoT permite la transmisión de datos de hasta 1600 bytes. LoRa permite enviar un máximo de 243 bytes de datos. Por el contrario, Sigfox propone la longitud de carga útil más baja de 12 bytes, lo que limita su utilización en varias aplicaciones de IoT que necesitan enviar grandes tamaños de datos.

Para el presente caso los datos que se trabajaran en la red sera de decenas de bytes con unos 6 dispositivos que conformaran la misma, por lo cual seria LoRa la mejor elección para la red a implementar.

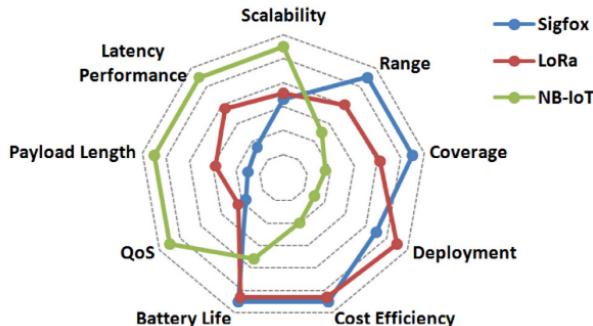


Figura 5.3: Comparación entre Sigfox, LoRa y NB-IoT en términos de factores de IoT

Fuente: ([Mekki et al., 2019](#))

### 5.2.6. DISPOSITIVO SELECCIONADO

En base a la comparación realizada sobre los diferentes factores de IoT, se implementara la red de sensores con módulos LoRa ya que las características que presenta se acomodan mejor al problema que se quiere solucionar para lo cual se realizo una búsqueda sobre diferentes dispositivos que aplican esta tecnología.

Se selecciono el modulo "Semtech SX1272 LoRa Mbed Enabled Shield" que se muestra en la figura 5.4, Los transceptores SX1272 / 73 cuentan con el módem de largo alcance LoRaTM que proporciona comunicación de espectro ensanchado de alcance ultra-largo y alta inmunidad a interferencias al tiempo que minimiza el consumo de corriente. El uso de la técnica de modulación LoRaTM patentada de Semtech SX1272 / 73 puede lograr una sensibilidad de más de -137 dBm utilizando un cristal de bajo costo y una lista de materiales. La alta sensibilidad combinada con el amplificador de potencia integrado de +20 dBm produce un presupuesto de enlace líder en la industria que lo hace óptimo para cualquier aplicación que requiera rango o robustez. LoRaTM también ofrece ventajas significativas tanto en el bloqueo como en la selectividad sobre las técnicas de modulación convencionales, resolviendo el compromiso de diseño tradicional entre rango, inmunidad a interferencias y consumo de energía ([mbed, 2019](#))..

5

- Presupuesto de enlace máximo de 157 dB
- Salida de RF constante de +20 dBm - 100 mW
- Velocidad de bits programable hasta 300 kbps.
- Alta sensibilidad: hasta -137 dBm
- Parte frontal a prueba de balas: IIP3 = -12.5 dBm
- 89 dB bloqueando la inmunidad
- Corriente de RX baja de 10 mA, retención de registro de 200 nA
- Sintetizador totalmente integrado con una resolución de 61 Hz.
- Modulaciones FSK, GFSK, MSK, GMSK, LORA y OOK
- Sincronizador de bits incorporado para la recuperación del reloj
- Sincronización de reconocimiento de palabras
- Detección de preámbulos
- 127 dB + rango dinámico RSSI
- Sentido RF automático con AFC ultrarrápido
- Motor de paquetes de hasta 64 bytes con CRC.

Con estos dispositivos se realizo la implementación el los nodos para la red como se muestran en la figura 5.7 en la parte superior de la imagen se observa en modulo de comunicaciones Semtech SX1272 seguido de este un puente para realizar la conexión del la raspberry pi zero wireless.

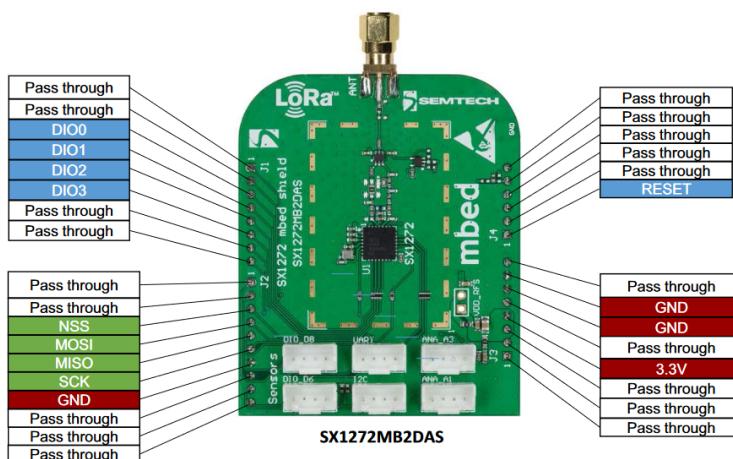


Figura 5.4: Semtech SX1272 LoRa Mbed Enabled Shield

Fuente: ([mbed](#), 2019)

### 5.3. DISEÑO E IMPLEMENTACIÓN DE LA RED

En base a la hipótesis del proyecto a al modelo del nodo mostrado en la figura 5.5 se estableció una arquitectura de red como se muestra a continuación:

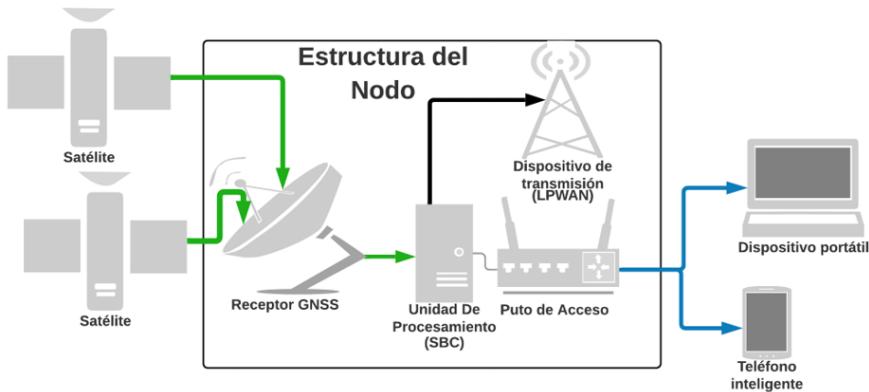


Figura 5.5: Conexiones de un Nodo de la red

Fuente: Propia

#### 5.3.1. ARQUITECTURA DE RED

El protocolo de comunicación handshaking de enlace es un proceso automatizado que establece parámetros para la comunicación entre dos dispositivos diferentes antes de que comience la comunicación normal. Al igual que la forma en que un apretón de manos humano prepara el escenario para que la comunicación siga, el apretón de manos informático proporciona a ambos dispositivos las reglas básicas para la forma en que los datos deben compartirse entre ellos. Estas reglas pueden incluir velocidad de transferencia, alfabeto de codificación, paridad, procedimiento de interrupción y más.

Una vez que se completa el protocolo de handshake, los datos de la aplicación pueden comenzar a fluir entre un cliente y el servidor. El cliente puede enviar un paquete de datos inmediatamente después del paquete ACK, y el servidor debe esperar el ACK antes de que pueda enviar cualquier dato. Este proceso de inicio se aplica a todas las conexiones TCP y conlleva una implicación importante para el rendimiento de todas las aplicaciones de red que utilizan TCP: cada nueva conexión tendrá una vuelta completa de latencia antes de que se puedan transferir los datos de la aplicación.

La figura 5.6 muestra esquemáticamente la red inalámbrica propuesta para los experimentos donde cada nodo M,S posee los elementos mostrados en la figura 5.5, en el lado izquierdo de la imagen se muestra un estado inicial de la red en la cual el nodo maestro (M) se encarga de establecer una comunicación por medio del protocolo handshaking con el primer nodo esclavo ubicado su izquierda que sería el camino 1 presentado en la imagen donde se establece la comunicación, posteriormente el nodo S regresa la información que tenga disponible para enviar al nodo M, cuando M recibe la información del

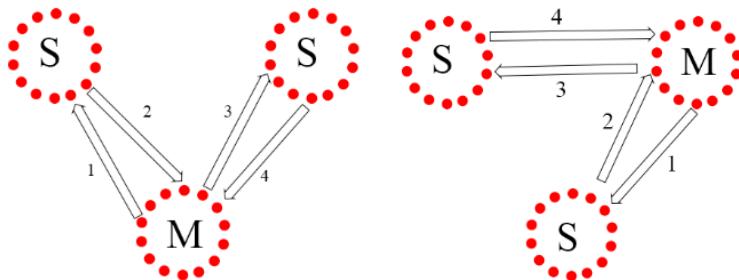


Figura 5.6: Topología de la red de comunicaciones

Fuente: Propia

5

primer nodo S da por termina esta comunicación y pasa a preguntar al siguiente nodo en la red que para este caso sería el nodo S ubicado en la izquierda.

Cuando el nodo M pregunta al nodo S en el lado derecho este le pide ser el nuevo maestro de la red lo cual nos lleva a un segundo estado de la red en el cual el nodo M cambia, y este nuevo nodo M realiza la misma acción de establecer comunicación basada en handshaking, esto permite que cualquier nodo de la red se pueda comportar como un punto central de la red donde se concentra toda la información.

### 5.3.2. IMPLEMENTACIÓN RED

Para la conexión del modulo SX1272 ([mbed, 2019](#)) con la placa Raspberry Pi Zero Wireless ([Raspberry, 2018](#)) se realizo por medio de un adaptador lo que permite un control directo por medio de la librería SX1272 diseñada para arduino, por lo cual se requiere implementar la arduPi en la raspberry pi que permite utilizar librerías de arduino en base a la documentación ([Hacks, 2012](#)).

Se propone que cada nodo de la red pueda ser un punto central que designe desde qué nodo desea solicitar la información mediante handshaking, estableciendo comunicación M2M entre los nodos M y S, de tal forma que toda la información pueda llegar a un punto, para establecer el cambio de M en la red se simulo la conexión de un usuario que solicita desde un nodo S el control de la red.

La topología de la red se basará en la comunicación M2M donde un nodo central M se encargará de establecer las comunicaciones en la red, generando así una red como se muestra en la figura 5.6, para hacer la red se utilizó el módulo SX1272, que se conecta mediante una puente con la computadora de placa reducida raspberry pi zero wireless, ya que el módulo SX1272 tiene conexiones I2C y SPI entre otros de fácil acceso en caso de requerir conectar sensores ya sea para medir temperatura, presión posición, etc. o el control de dispositivos en el lugar del nodo, la red implementada la finalizar el proyecto se presenta en la imagen 5.8.

Para establecer la comunicación entre los nodos su utilizo una librería diseñada para el módulo SX1272, en la cual se permiten variar los parámetros vistos en las ecuaciones 3.5 y 3.6, para lo cual se establecieron los siguientes parámetros: una frecuencia de portadora de 915 MHz, BW de 500 KHz y CR de 4/5.

Para lo comunicación al momento de que el nodo M realiza pregunta sobre la existencia del nodo por medio de una confirmación en forma de ACK que verifica el funcionamiento del nodo S, ya establecida una comunicación entre los dos nodos S envía la información , de igual manera el nodo M confirma el recibir la información por medio de un ACK, la información que se envió para el correspondía al estado del módulo SX1272 siendo paquetes enviados, paquetes recibidos y tiempos de envío de los paquetes desde que el nodo M pregunta hasta que regresa el ACK al nodo S , el nodo M recibe esta información y lo guarda en un archivo correspondiente al nodo que se usara para un posterior análisis.

## 5

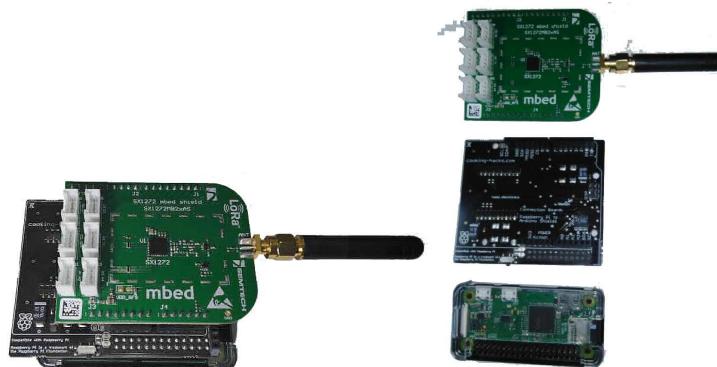


Figura 5.7: Nodo de la Red (Raspberry pi Zero + sx1272 + Bridge)  
Fuente: Propia

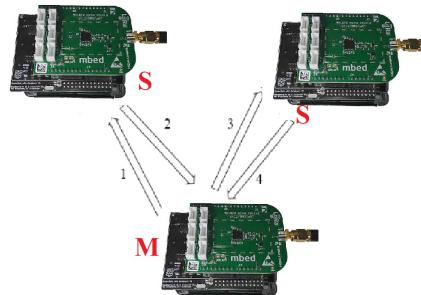


Figura 5.8: Modelo Final de la red Con nodos Reales  
Fuente: Propia

## 5.4. DISEÑO DE INTERFAZ GRÁFICA

Para el desarrollo se estableció del proyecto se estableció un entorno gráfico final que se requería para la visualización de la información , teniendo en cuenta que todo el proyecto de manejará de manera offline, por lo cual cada nodo de la red tiene la capacidad de de generar una red wi-fi donde se desplegará la interfaz gráfica para la visualización de la información para lo cual se desarrollaron las siguientes vistas.

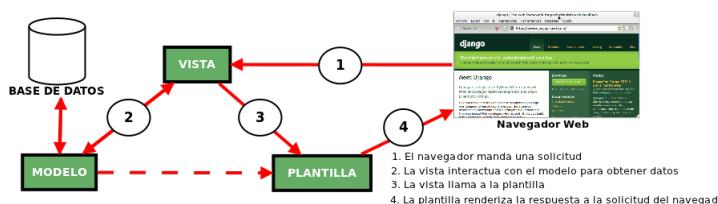


Figura 5.9: Modelo Funcionamiento django

Fuente Platzi

5

Como se observa en la figura 5.9 donde se muestra el funcionamiento de django, se basa en modelos que representan la base de datos , una vistas que se encargan de leer y editar los modelos para mostrarse por medio de plantilla, a continuación se muestra que se realizó en cada uno de estos pasos:

### 5.4.1. MODELOS

Para implementar la interfaz gráfica en django se diseño primero los modelos utilizados para el control de archivos generados como se muestra en la tabla 5.2 , para saber si el nodo actual es el nodo maestro se utilizó un modelo presentado en la tabla 5.4 y para visualización de la información se utilizó el modelo presentado en la tabla 5.4 donde se establecen los principales indicadores a mostrar en la aplicación.

Lista	
Id	AutoField
archivo	FileField
nombre	TextField
size	integerField

Cuadro 5.2: Modelo De lista de Archivos

maestroEs	
Id	AutoField
maestro	BooleanField

Cuadro 5.3: Modelo Maestro-Esclavo

Nodos	
Id	AutoField
AnchoBanda	IntegerField
CargaUtil	TextField
Corriente	IntegerField
EstadoLora	BooleanField
FuerzaSenal	IntegerField
NumeroNodo	IntegerField
NumeroSatelites	IntegerField
PaquetesEnviados	IntegerField
PaquetesRecibidos	IntegerField
TiempoEnvio	FloatField
altitud	FloatField
dilucion	FloatField
estadoGnss	BooleanField
fixQuality	IntegerField
latitud	FloatField
longitud	FloatField

Cuadro 5.4: Modelo De Nodos

5

#### 5.4.2. VISTAS

Para django una vista , son las distintas funcionalidad que puede tener la aplicación que se muestra sobre un plantilla, a continuación se muestran las diferentes vistas implementadas y una breve descripción de cada una:

- **download:** Donde permite a un usuario descargar la información de un nodo.
- **ver:** Donde permite a un usuario Visualizar la información de un nodo.
- **maestro:** Donde se realiza cambios sobre el modelo de maestroEs pasando el nodo a ser el maestro o esclavo de la red.
- **actualizar:** Donde se actualiza el modelo de nodos de django con la información que tiene la raspberry sobre toda la red.
- **detallesNodo:** Donde se muestra en un tabla de detalles actuales de un nodo.
- **index:** Donde se lleva al usuario a la pagina principal y muestra una mínimo de información desde el modelo de nodos.

#### 5.4.3. TEMPLATES

Para el diseño de la aplicación se tiene en cuenta las diferentes vistas planteadas para lo cual se obtienen las siguientes :

- **Zona Wi-Fi disponibles** Por medio de la raspberry pi zero wireless se genera un punto de acceso basado en IEEE 802.11ac con contraseña lora como se muestra en la figura 5.10 (a).

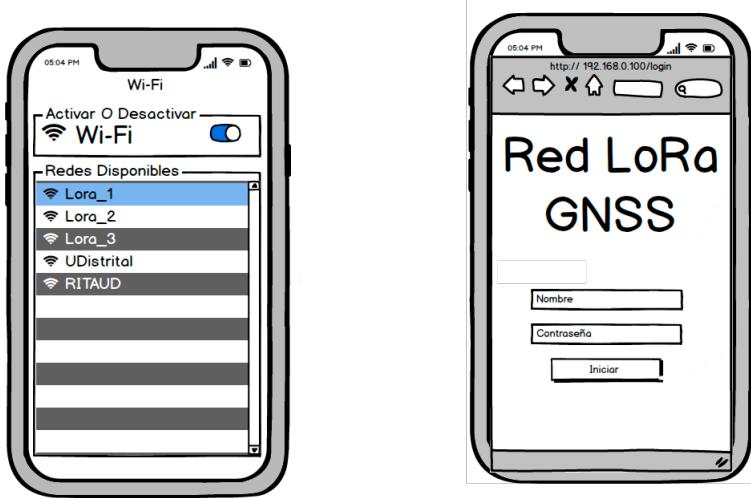


Figura 5.10: a) Vista de redes disponibles b) Inicio Sesión

Fuente:Propia

5

- Inicio de Sesión Se implementa un login al sistema, para tener un control de usuarios, donde el usuario u contraseña son "loraçomo se muestra en la figura 5.10 (b).

- **Lista De Nodos:**

Se tiene una primera vista donde se visualizan los nodos disponibles en la red y su estado visto desde el nodo como se muestra en la figura 5.11, contara con un botón de actualizar, donde se recargar la información correspondiente a la red desde el nodo en cuestión.

- **Detalles de Nodo**

Al seleccionar un nodo en la vista de lista de nodos permite al usuario ir a una nueva vista donde se observaran datos específicos del nodo seleccionado respecto a el estado de modulo de comunicaciones como a la información recogida por este desde el dispositivo GNSS como se muestra en la figura 5.12 así como un botón para descargar la información de los nodos a cualquier dispositivo.

Por medio los los modulo lora ubicados en cada modulo se trasmite la información de cada uno de los nodos tanto del estado general (GNSS y LoRa) como la información obtenida por el modulo GNSS.

Para el estado del GNSS se tuvieron en cuenta los siguientes indicadores tomando el ultimo punto de actualizado:

- Estado (Toma Información)
- Numero de satélites.
- Dilución horizontal de la posición.
- Latitud
- Longitud

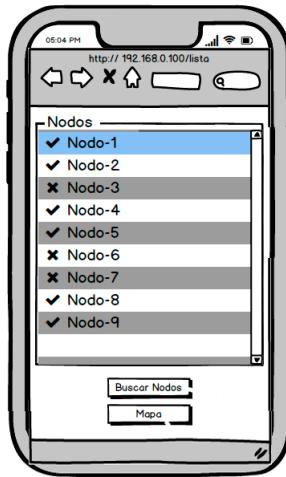


Figura 5.11: Lista de Nodos

Fuente: Propia

5

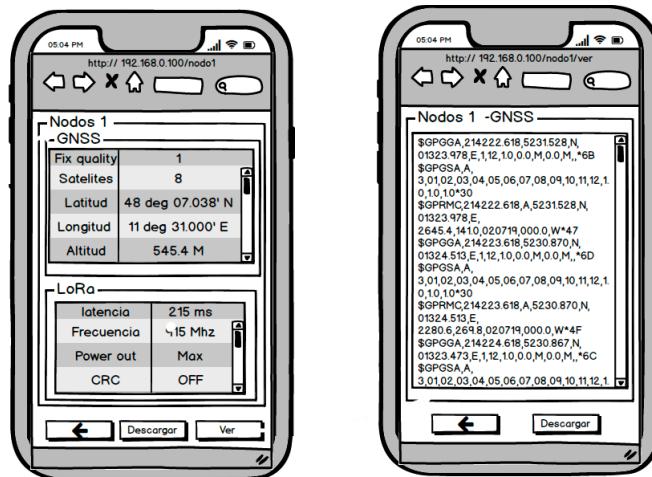


Figura 5.12: Detalles de Nodos

Fuente: Propia

- Altitud
- Fix Quality
  - 0 : Invalido
  - 1 : GPS fix (SPS)
  - 2 : DGPS fix
  - 3 : PPS fix
  - 4 : Real Time Kinematic

En cuanto a modulo de comunican LoRa se tomaron los siguientes:

- Estado Habilitado/Des-Habilitado
- Numero Nodo
- Ancho de Banda.
- Canal.
- Corriente.
- Temperatura.
- Paquetes Enviado.
- Paquetes Recibido.
- Tiempo De Envío.
- RSSI (Fuerza Señal) .
- Payload (Carga Útil).

#### 5.4.4. LENGUAJE PROGRAMACIÓN Y FRAMEWORK

Para el desarrollo de toda la aplicación tanto de la parte visual como del control de la red se busco un Framework basado en python, ya que este presenta facilidad en el procesamiento de datos como en la implementación de diferentes algoritmos.

5

En la figura 5.13 se muestra una comparación entre Python, PHP, Node.js en la respuesta que poseen estos entornos en el despliegue de un "Hola Mundo" en varios usuarios en la primera donde python presenta una mejor respuesta que PHP y node.js con menos de 200 usuarios, para nuestro caso los usuarios serian la cantidad de nodos que posee la red por lo cual python es la mejor opción ya que se trabajara con un máximo de 8 nodos .

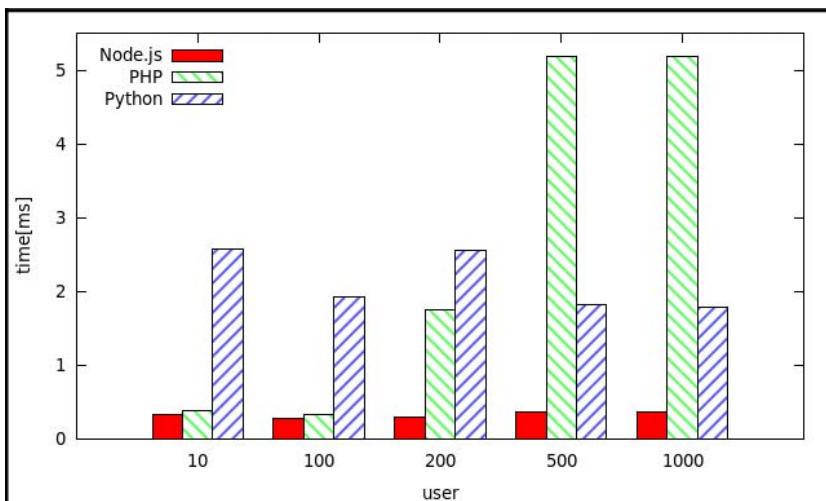


Figura 5.13: Comparación Python,PHP,Node.js server response  
Fuente: [Lei et al. \(2014\)](#)

Para el caso de procesamiento de datos se ilustra en la figura 5.14 donde se compara

con los primeros diez términos de la serie de fibonacci ejecutados por Python Node.js y PHP dando python presenta una mejor respuesta con menos de 1000 usuarios.

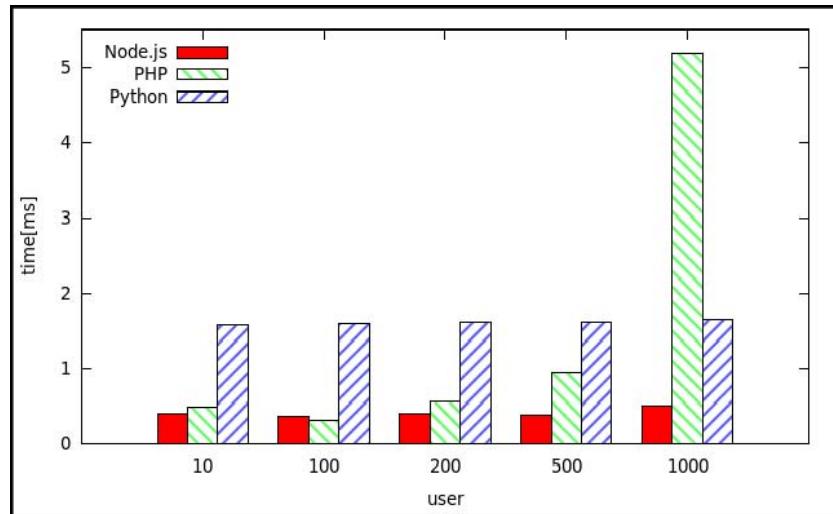


Figura 5.14: Comparación Python,PHP,Node.js Serie de Fibonacci  
Fuente: [Lei et al. \(2014\)](#)

Un framework es un marco de trabajo. En programación hace referencia a una serie de herramientas con las que puedes construir algo más fácil y rápido con algún lenguaje de programación.

Para el desarrollo del entorno gráfico se utilizará el framework de Django que es basado en python.

Django es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como Modelo–vista–template. Fue desarrollado en origen para gestionar varias páginas orientadas a noticias de la World Company de Lawrence, Kansas, y fue liberada al público bajo una licencia BSD en julio de 2005; el framework fue nombrado en alusión al guitarrista de jazz gitano Django Reinhardt. En junio de 2008 fue anunciado que la recién formada Django Software Foundation se haría cargo de Django en el futuro.

La meta fundamental de DJANGO es facilitar la creación de sitios web complejos. DJANGO pone énfasis en el re-uso, la conectividad y extensibilidad de componentes, el desarrollo rápido y el principio No te repitas (DRY, del inglés Don't Repeat Yourself). Python es usado en todas las partes del framework, incluso en configuraciones,archivos, y en los modelos de datos.

#### 5.4.5. IMPLEMENTACIÓN

Por medio del framework django basada en un platilla se generaron las siguientes vistas para la interfaz gráfica, en la figura 5.15 se muestra el inicio de sesión y la lista

de nodos, en la lista de nodos se puede ver y actualizar el estado así como establecer la configuración del nodo para que se comporte como en nodo maestro en la red.

The screenshot shows a web application interface for managing nodes. At the top, there is a login form titled "Iniciar sesión" with fields for "Nombre" (lora) and "Contraseña" (\*\*\*\*). There is also a "Recordar contraseña" checkbox and a "Iniciar sesión" button. Below the login is a link "¿Se te olvidó tu contraseña?".

Below the login, the URL "Nodos /" is visible. The main content area displays a table with two columns: "Estado" and "Nodos". The table contains three rows, each with a checkmark icon in the "Estado" column and a node name in the "Nodos" column: "Nodo 1", "Nodo 3", and "Nodo 5".

At the bottom, there is a map titled "Actualizar" showing a park area with several blue location markers. The map includes labels for "Avenida Carrera 59A", "Transversal 59A", "Calle 60", "Carrera 50A", "Parque Virgilio Barco", and "AC 63". A Mapbox logo is in the bottom left corner, and a copyright notice "© Mapbox © OpenStreetMap Improve this map" is in the bottom right corner.

Figura 5.15: Detalles de Nodo Django  
Fuente: Propia

En la figura 5.16 se observan los datos que pertenecen en este caso para el nodo 2, donde en la parte superior se observa la información perteneciente al modulo de comunicaciones LPWAN como se estableció anteriormente y en la parte inferior de la imagen se muestra los datos del sensor GNSS del nodo, como tambien se observan dos botones uno para descargar los datos del receptor GNSS y otro para regresar a la vista de lista.

5

LoRa	
Nombre	Estado
Estado	✓
Numero de Nodo	5
Ancho de Banda	0
Correinte	240
Paquetes Enviados	40
Paquetes Recividos	40
Tiempo Envio	0.0
RSSI(Fuerza Señal)	-104
Payload (Carga Util	255

GNSS	
Nombre	Estado
Estado	✓
Numero Satelites	12
Dilucion Horizontal de la Posicion	0.65
Latitud	4.6348666666666665
Longitud	-74.06821616666667
Altitud	2588.9
Fix Quality	1

[Descargar](#)

[Ver Datos](#)

[Regresar](#)

Figura 5.16: Detalles de Nodo Django  
Fuente: Propia

## 5.5. PRUEBAS DE CAMPO

En esta sección se muestran los experimentos realizados sobre la red para generar una caracterización de la misma , así como los resultados obtenidos en los diferentes escenarios, para lo cual primero se realizaron pruebas sobre la comunicación de los dispositivos LPWAN para verificar su funcionamiento en diferentes anchos de banda(BW) y SF, posteriormente se realizo una validación sobre la precesión de los GNSS tomando uno como base y los otros 2 como robbers, posteriormente se integraron estas dos para en un entorno rural realizar pruebas del funcionamiento de la red incorporando tanto la recolección de datos del GNSS como su trasmisión con el dispositivo LoRa .

### 5.5.1. LPWAN

Se realizaron pruebas sobre el funcionamiento de los sensores para lo cual se fijo una distancia fija entre los nodos de 5 m para verificar el funcionamiento de la red, variando el SF de la red entre 7 y 9 como tambien se vario el ancho de banda disponible para la red entre 125 Hz y 250 Hz.

Para lo cual se tomaron tres nodos un Maestro y dos esclavos donde se envió información de los nodos esclavos (1 y 5) hacia el nodo maestro 3 manteniendo la distancia constante ente los puntos.

En las figura 5.17 y 5.18 se ilustra los tiempos promedio para el envío de paquetes con un SF de 7 y 9 en un total de 100 mensajes, como se observa para el caso de un SF de 9 con una ancho de banda de 125 Hz presento un tiempo promedio de 6.911 Seg en contraste con un ancho de banda de 250Hz se presenta un tiempo promedio de 2.369 segundos.

Para el caso de la figura 5.18 se presentan tiempos en la trasmisión menores al caso de SF = 9, para el caso de 125Hz un tiempo promedio de 3.45 segundos, y para el caso de 250Hz se tiene un tiempo promedio de 2.1 segundos en los 100 paquetes enviado.

El código implementado para realizar las configuración de los nodos se encuentra en el anexo 4 .

En las figuras 5.19 y 5.20 se muestran los porcentajes de paquetes perdidos en una trasmisión de 100 paquetes ,para el caso de SF de 7 se presentan errores mayores al 10% en contraste con un SF de 9 no se presentaron errores mayores al 5% en la trasmisión de datos.

### 5.5.2. GNSS

Para verificar el funcionamiento del sensor GNSS para lo cual se tomaron datos por medio de la raspberry pi zero w, para lo cual se utilizo el siguiente código, en el cual se en primero se establece la conexión con el puerto USB, a una baudrate de 9600 baudios, y una tamaño de bits de 8.

En primera instancia se tomaron los datos en formato nmea para realizar el envío de datos entre los nodos,de tal manera que se tuviera una mínimo de información sobre el

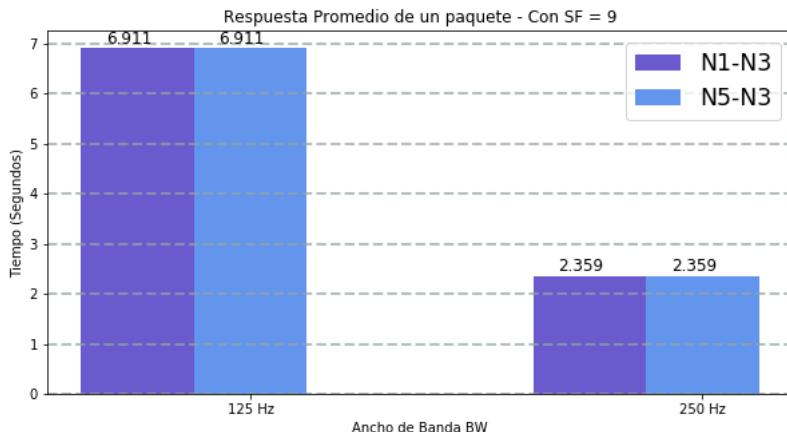


Figura 5.17: Tiempos de envío promedio con SF 9  
Fuente: Propia

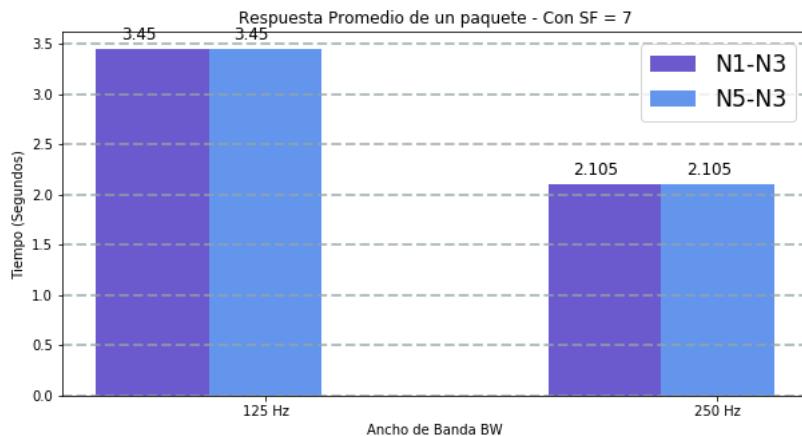


Figura 5.18: Tiempos de envío promedio con SF 7  
Fuente: Propia

posicionamiento de los nodos, a continuación se presenta el código y formato de datos NMEA.

```

1 import serial
2 import time, os
3 import pynmea2
4 import sys
5 import os
6 port = serial.Serial(port="/dev/ttyACM0",
7                         baudrate=9600,
```

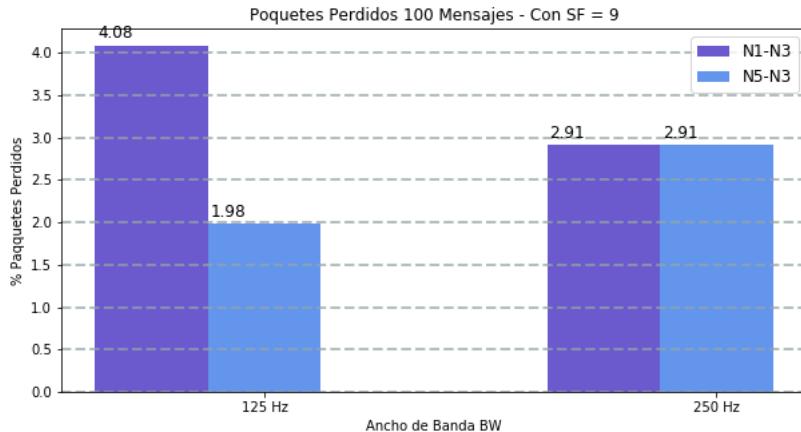


Figura 5.19: Paquetes Perdidos SF 9  
Fuente: Propia

5

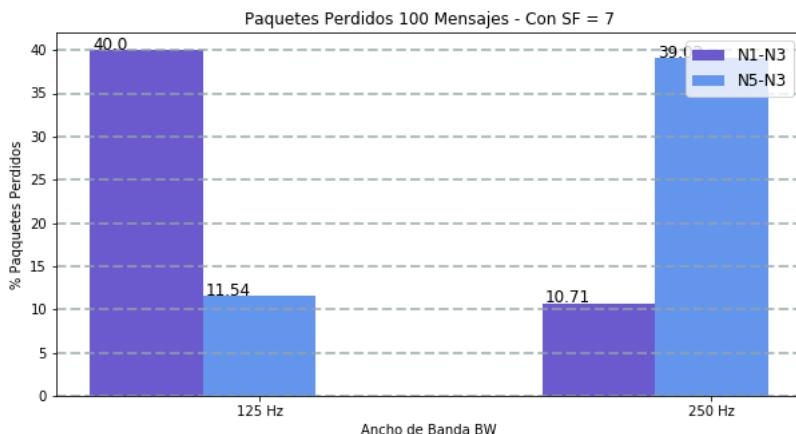


Figura 5.20: Paquetes Perdidos SF 7  
Fuente: Propia

```

8           timeout=0.1,
9           bytesize=serial.EIGHTBITS)
10
11 fi=open("/home/pi//Desktop//nodo.txt", "a+")
12
13 port.close()
14 port.open()
15 while True:
16     data = port.readline()

```

```

17     if (data.startswith("$GNGGA")):
18         msg = pynmea2.parse(data)
19         fi.write(data)
20         print(data)

1 $GNGGA ,001914.00 ,0438.09531 ,N ,07404.10232 ,W ,1 ,08 ,1.62 ,2587.1 ,M ,4.7 ,M
2 ,,*5E
3 $GNGGA ,001915.00 ,0438.09532 ,N ,07404.10232 ,W ,1 ,08 ,1.62 ,2586.9 ,M ,4.7 ,M
4 ,,*55
5 $GNGGA ,001916.00 ,0438.09532 ,N ,07404.10233 ,W ,1 ,08 ,1.62 ,2586.7 ,M ,4.7 ,M
6 ,,*59

```

Para la correcta georreferenciación de un los puntos se utilizaron data en formato RAW que permite una mejor aproximación de la ubicación real de los nodos en la figura 5.21 se ilustra la información de dos nodos posterior al procesamiento requeridos para la corrección de los datos respecto a un punto central tomado, utilizando la librería RTKLIB, lo configuración necesaria para toma de datos en formato RAW desde el modulo F9P se especifica en el anexo 2.

## 5

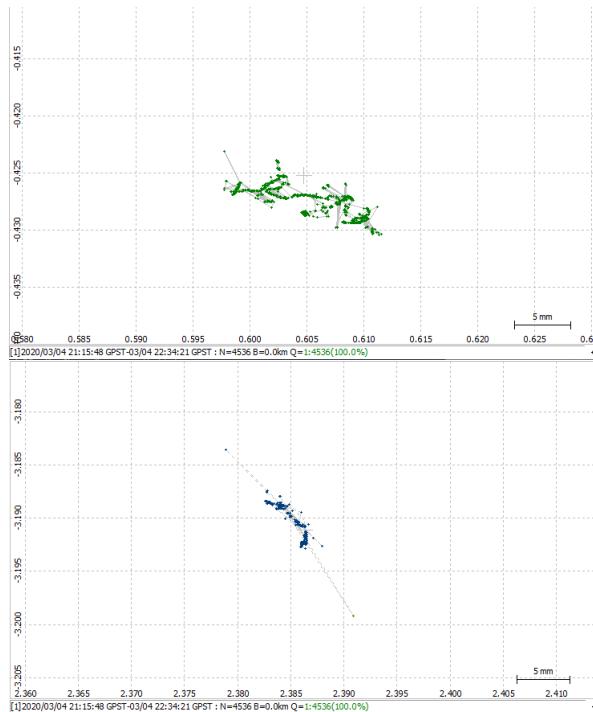


Figura 5.21: Posicionamiento de puntos  
Fuente: Propia

En base a los datos corregidos obtenidos se obtiene los variación en los datos en la imagen 5.22 se observa el diagrama de cajas en respecto al eje X en uno de los nodos, teniendo indicadores tales como:

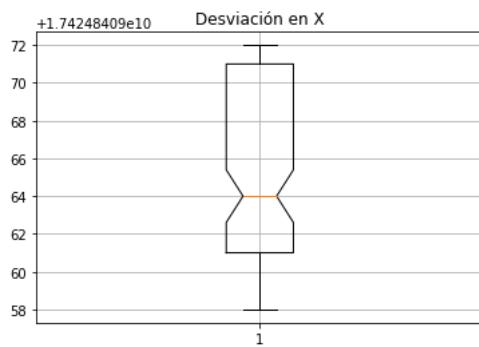


Figura 5.22: Desviación En X  
Fuente: Propia

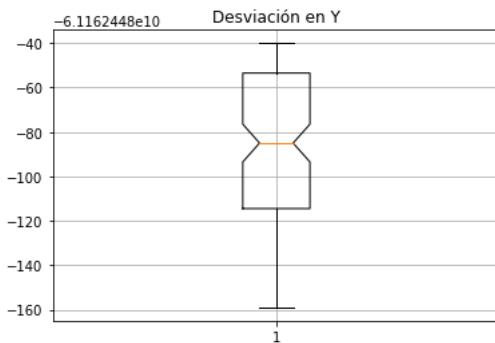


Figura 5.23: Desviación En Y  
Fuente: Propia

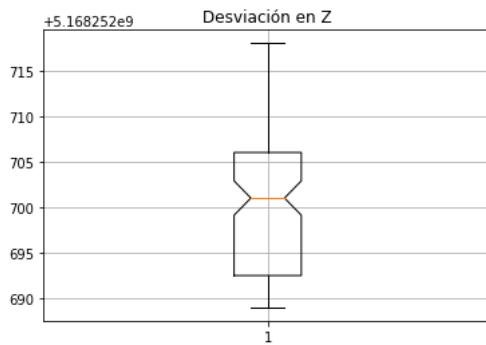


Figura 5.24: Desviación En Z  
Fuente: Propia

- Media: 17424840965
- Mediana: 17424840964
- Moda: 17424840971
- Desviación : 4.568262

para el caso del el posicionamiento en el eje Y como se ilustra en la figura 5.23 obteniendo indicadores:

- Media: -61162448090
- Mediana: -61162448085
- Moda: -61162448051
- Desviación : 37.243805

para el caso del el posicionamiento de altura (eje Z ) como se ilustra en la figura 5.24 obteniendo indicadores :

- Media: 5168252700
- Mediana: 5168252701
- Moda: 5168252693
- Desviación : 8.212693

5

### 5.5.3. RED DE SENSORES

Se plantearon dos experimentos para observar la funcionalidad de la red, en los cuales se rotará entre los dos de los tres nodos disponibles en la red de manera autónomo el nodo maestro de esta, dejando un nodo siempre fijo en modo esclavo simulando así la interacción de un usuario sobre los nodos esta variación se realizará cada 50 paquetes de información enviados, en la figura 5.25 se presenta la imagen de un nodo utilizado en las pruebas, ya con el modulo GNSS y la batería pasa su funcionamiento.

En el primer experimento se tomó un SF de 7 teniendo así una red con tiempos al aire cortos y un corto alcance de la red, como se observa en la figura 6 el nodo 1 y 2 del a red se encuentra a una distancia aproximada de 100 m, y se ubica un tercer nodo a una distancia inicial de 200 m y en un segundo momento a 400 m de los nodos 1 y 2 como se ilustra en la figura 5.26 en la parte derecha, se realizara un experimento similar cambiando el SF de la red a 9 para hacer una comparación entre estos dos resultados y como el ambiente afecta el desempeño de la red al modificar uno de sus parámetros.

Las pruebas se realizaron una un ambiente rural como se muestra en la figura 5.26, en la parte izquierda de la figura se aprecia la ubicación real de los nodos sobre el terreno, como se observa no hay una línea de visión directa entre los nodos en las posibles conexiones que se puedan presentar en la red, siempre se presenta obstrucción por los árboles en el camino.

### RESULTADOS

Para los resultados presentados se realizó la medición de los tiempos que se requieren para él envío de paquetes entre el nodo maestro(M) y esclavo(S), y la cantidad de paquetes recibidos y enviados por cada nodo, el tiempo empleado por el nodo maestro

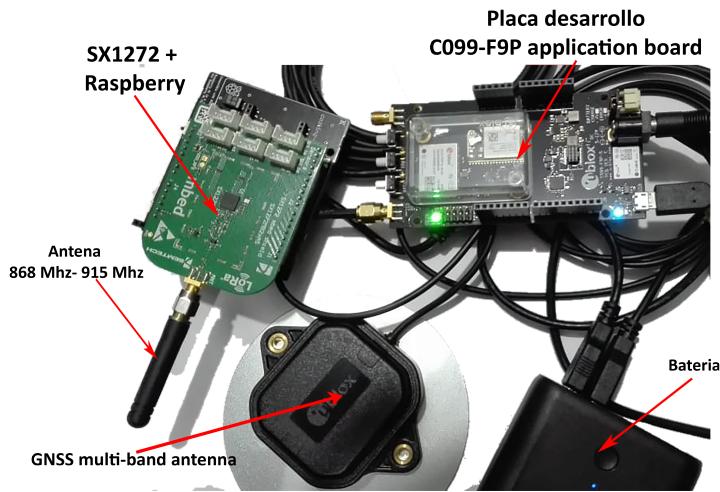


Figura 5.25: Vista real del nodo Implementado  
Fuente :Propia

5

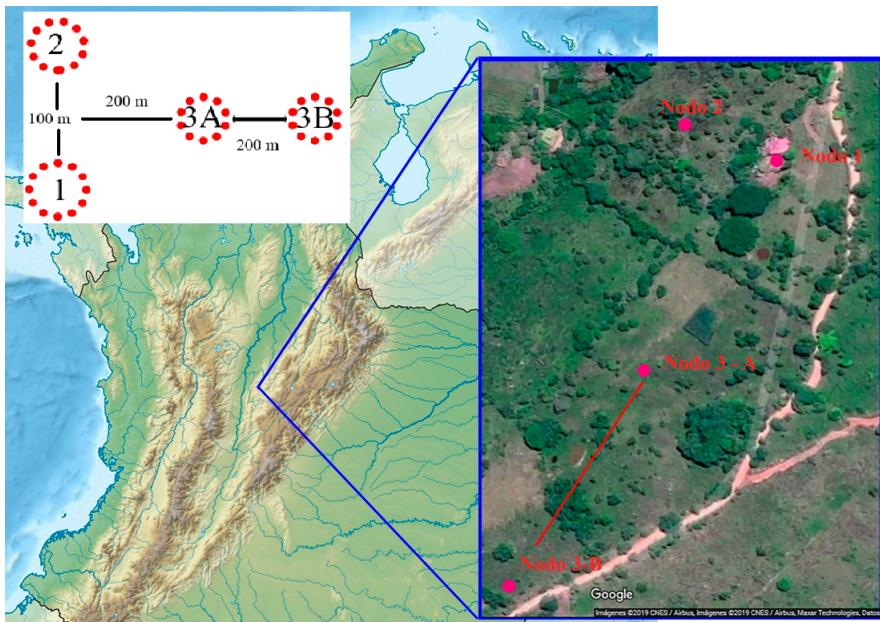


Figura 5.26: Experimento  
Fuente: Propia

para recorrer los otros nodos todos los nodos de la red, se tomó la cantidad de paquetes como la cantidad de veces que un nodo maestro logró establecer comunicación con un nodo esclavo y la cantidad de paquetes perdidos cuando el nodo esclavo no retorna la

información cuando ya se ha establecido la comunicación.

En las tablas 5.5 y 5.6 se presenta los datos obtenido con un SF de 7 y 9 donde se relacionan los tiempos y cantidad de paquetes manejados por los nodos, con una distancia el nodo 3 de 200 m, como se observa en las tablas 1 y 2 la comunicación entre los nodos 1 y 2 separados por 100 m no presentan problemas al momento de la trasmisión de paquetes, y los tiempos de trasmisión dese el momento que se establece la comunicación desde el nodo 1 hasta que finaliza con la comunicación con el nodo 2 permanece en un intervalo entre 2.132 segundos y 2.129 segundos sin presentarse perdida en los paquetes trasmitidos.

Los tiempos para el envío de paquetes no presentan grandes variaciones al momento de generar cambios en el SF manteniendo una diferencia menor a 1% en los tiempos, una situación similar ocurre con una distancia de 400 m donde no se presenta diferencia considerable donde no hay grandes diferencias de tiempos al realizar variación sobre SF, los tiempos que tomo el nodo maestro para recorrer los otros nodos de la red es de aproximadamente 4.49 segundos y 4.626 segundos, las variaciones en los tiempos para el envío de paquetes se ven afectados principalmente por árboles que se encuentran en el terreno, la perdida de paquetes es menor a un 10% de los paquetes enviados presentándose mayormente con la distancia del nodo 3 a 400 m, por la interferencia de los árboles y las altas temperaturas cercanas a los 30°C, que afecta el rendimiento general de los dispositivos.

Nodos	N1- N2		N1-N3		N3-N1		N3-N2	
SF	7	9	7	9	7	9	7	9
Distancia(M)	100		200		200		232	
Promedio Paquete (S)	2.121	2.132	2.350	2.352	2.355	2.352	2.352	2.355
Max Paquete (S)	2.209	2.209	2.355	2.357	2.359	2.356	2.357	2.360
Min Paquete (S)	2.092	2.092	2.348	2.348	2.347	2.348	2.348	2.347
Núm. paquetes	50	50	50	50	50	50	50	50
Núm. Paquetes Perdidos	0	0	1	2	3	2	2	3

Cuadro 5.5: Resultados con distancia 200 m  
Fuente : Propia

Nodos	N1- N2		N1-N3		N3-N1		N3-N2	
SF	7	9	7	9	7	9	7	9
Distancia(M)	100		400		400		412	
Promedio Paquete (S)	2.132	2.129	2.341	2.390	2.495	2.494	2.497	2.489
Max Paquete (S)	2.211	2.209	2.401	2.404	2.497	2.499	2.508	2.496
Min Paquete (S)	2.101	2.092	2.487	2.348	2.484	2.480	2.488	2.486
Núm. paquetes	50	50	50	50	50	50	50	50
Núm. Paquetes Perdidos	0	0	2	4	3	4	1	3

Cuadro 5.6: Resultados con distancia 400 m  
Fuente : Propia

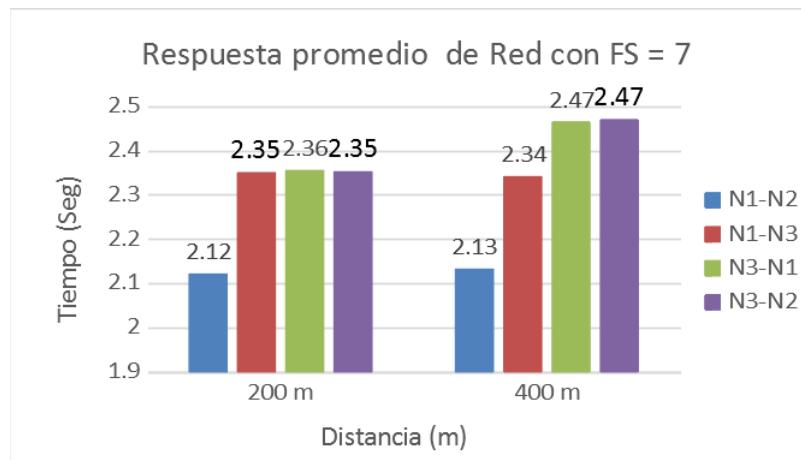


Figura 5.27: Respuesta de la red con SF de 7 en 200 m y 400 m

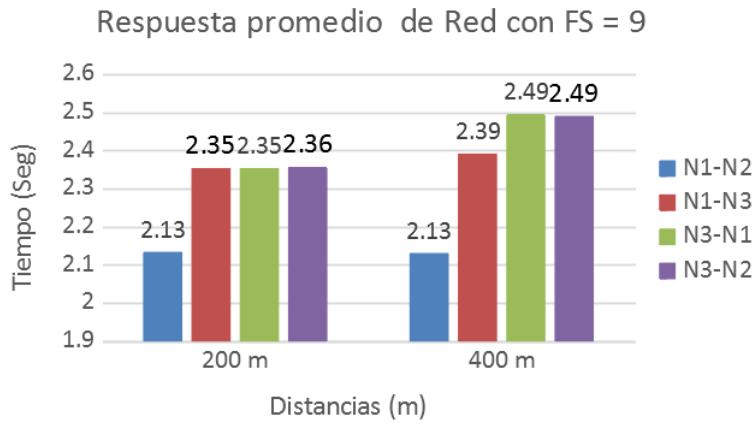
Fuente: Propia

5

En las Figuras 5.27 y 5.28 se presenta los datos obtenido con un SF de 7 y SF de 9 donde se varia las distancias del nodo 3 entre 200 m y 400 m, observando los tiempos empleados para la trasmisión de datos entre dos nodos M, S. Para el caso de SF de 7 como se muestra en la figura 5.27, en el caso donde el nodo maestro es N1 se presenta tiempos de trasmisión con el nodo N2 cercanos a 2.121 segundos con una variación no mayor a 0.011 segundos, el intercambio de paquetes que se presenta entre los nodos 1 y 3 en una distancia de 200 m no presenta cambios significativos en los tiempos para el envío de información cercanos a 0.005 segundos, en cambio para la distancia de 400 m se presentó una diferencia de tiempo 5,29% en promedio.

En la figura 5.28 se presenta un comportamiento muy similar al descrito para la figura 5.27, en 200 m no se presentan variaciones de tiempo apreciables, para 400 m se presenta una variación en los tiempos requeridos por la red para el envío de paquetes principalmente cuando se interactúa con el nodo 3 de la red presentando variaciones de 1.16% respecto al presentado con un SF de 7.

En la figura 5.29 se presentan los paquetes perdido en la trasmisión de datos, en la comunicación ente los nodos 1 y 2 no se presentó una pérdida de paquetes, con un SF de 9 y una distancia de 400 m se presenta un error del 8 % de los paquetes que se envían entre los nodos 1 , 3 y presentándose un error de 6% entre los nodos 2 , 3 para esta distancia, siendo este experimento el que presento mayores porcentajes de error en la trasmisión de datos, para la distancia de 200 m se presenta la menor cantidad de errores con cualquiera de los dos SF utilizados para el experimentos, siendo un SF de 9 donde hay mayor posibilidad de que se presente un error en la comunicación .



5

Figura 5.28: Respuesta de la red con SF de 9 en 200 m y 400 m  
Fuente: Propia

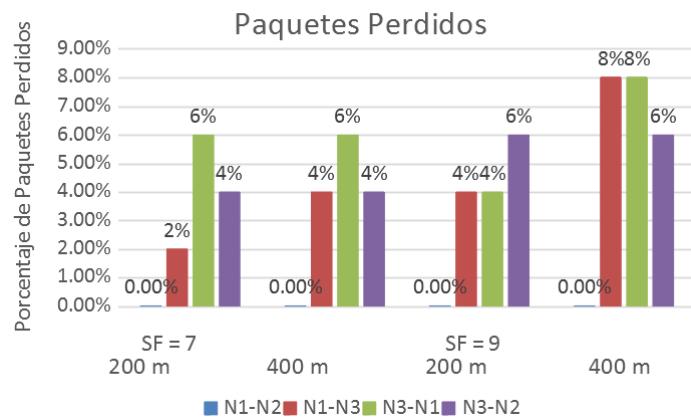


Figura 5.29: Porcentaje de Paquetes Perdidos  
Fuente: Propia

# 6

## CONCLUSIONES

- En la actualidad los dispositivos GNSS son ampliamente utilizados en diversos campos con diversas aplicaciones, para este caso en específico se requería que además de ser preciso y exacto el gnss tuviera la capacidad de enviar datos en crudo,
- Hay varios dispositivos en el mercado que permiten implementar redes LPWAN, con varios fabricantes y diversas características como lo son las distancias máximas de trasmisión, la cantidad de paquetes, etc
- Se realizo un proyecto en django que nos permite visualizar a manera de tablas o en mapa, la información que es recolectada por la red.
- Como se observan en esta topología de red se presentan variaciones en la cantidad de perdida de paquetes y tiempos de la red al tener variaciones en el SF, al aumenta el SF se aumenta el tiempo al aire de trasmisión, así como la posibilidad de error en el envío de paquetes, como se observó en la figura 5.29al tener una mayor SF aumenta el tiempo en el aire del mensaje lo que hace mas vulnerable la red a interferencia generando un mayor porcentaje de error.
- Al implementar esta topología de red permite un usuario que se conecte a la red desde cualquier punto y para obtener la información de toda la red basados en que los módulos utilizados no soportan comunicación full - duplex, en base a los datos adquiridos se observa que es posible implementar este tipo de red teniendo tiempos de respuesta de algunos segundos, esto no presenta problema ya que la mayoría de los sensores no presentan grandes variaciones en los valores.
- A medida que se aumenten los nodos de la red se aumentara en una proporción de igual magnitud el tiempo requerido de un nodo Maestro para recibir toda la información de la red, los errores que se pueden presentar en la comunicación de la red están estrechamente relacionados con el medio ambiente en el que se desarrollan las pruebas, y la distancias entre los nodos, ya que como observo en

los datos a una mayor distancia se presentaron un mayor porcentaje de error en la trasmisión de paquetes.

- Al variar el SF de trasmisión de la red manteniendo las mismas distancias entre los nodos, se presentan variaciones entre los tiempos necesarios para el envío de la información, como se observa con un SF de 9 los tiempos promedio aumentan con respecto a los tiempos promedio de un SF de 7.
- Uno de los factores que afectan la perdida de paquetes en la transmisión son obstáculos presentes en el terreno, como lo pueden ser la presencia de arboles, edificios o señales de radio que generar interferencia.

# 7

## TRABAJOS FUTUROS

- Ampliar la cobertura de los módulos aprovechando su capacidad con una separación en Km, ya que en las pruebas realizadas no se utilizó la máxima distancia permitida por los módulos LoRa.
- Implementar la red con módulos que soporten full-duplex, ya que para este caso los módulos proporcionados por el grupo de investigación GITUD son half-duplex por lo cual para establecer la comunicación entre los nodos de tal forma que enviaran y recibieran datos se realizan dos enlaces.
- Agregar conexión con un gateway para implementar una red IoT, ya implementada la red LPWAN el siguiente paso es poder acceder a la información desde un Internet para lo cual se requiere establecer la configuración para establecer la conexión con un modulo gateway de LoRa.
- Implementar una arquitectura con más de 10 nodos en la red, LoRa una tecnología que puede soportar ciento de dispositivos conectados en un mismo instante de tiempo, por o cual se plantea aumentar la cantidad de nodos en la red para verificar su funcionamiento con redes mas robustas.
- Analizar precisión en modelos obtenidos con imágenes tomadas con drones, ya con los datos obtenidos por la red se y con fotos tomadas por drones se puede generar modelos de un terreno que permitan verificar el error de la red.
- Diseñar una interfaz que se actualiza automáticamente cada vez que se reciben datos de otros nodos, la interfaz gráfica desarrollada requiere que el usuario necesite actualizar "manualmente"los datos que se muestran, se plantearía actualizar la interfaz gráfica para que esta se actualice cada vez que la raspberry detecte un nuevo dato.



# BIBLIOGRAFÍA

- Aakvaag, N. and Frey, J.-E. (2006). Redes de sensores inalámbricos. *Revista ABB*, 2:39–42.
- Accuracy, G. (2013). National coordination office for space-based positioning, navigation, and timing.
- Aernouts, M., Berkvens, R., Van Vlaenderen, K., and Weyn, M. (2018). Sigfox and lorawan datasets for fingerprint localization in large urban and rural areas. *Data*, 3(2):13.
- Alliance, L. (2015). A technical overview of lora and lorawan. *White Paper, November*.
- Álvarez, D. A. G. (2008). Sistema gnss (global navigation satellite system). *Madrid: Universidad Autónoma de Madrid*.
- Bakuła, M., Przestrzelski, P., and Kaźmierczak, R. (2014). Reliable technology of centimeter gps/glonass surveying in forest environments. *IEEE Transactions on Geoscience and Remote Sensing*, 53(2):1029–1038.
- Bardyn, J.-P., Melly, T., Seller, O., and Sornin, N. (2016). Iot: The era of lpwan is starting now. In *ESSCIRC Conference 2016: 42nd European Solid-State Circuits Conference*, pages 25–30. IEEE.
- Castrillón, F. R. C. (2013). *Aplicaciones con minicomputadores raspberry pi provisto de módulo gps y acelerómetro para control de velocidad y posicionamiento*. PhD thesis, ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL.
- Commons, C. (2018). Decodinglora. url <https://revspace.nl/DecodingLora>.
- EFOR (2018). Tecnologías de comunicación para IoT. .
- Force, U. (2018). Gps accuracy. url <https://www.gps.gov/systems/gps/performance/accuracy/>.
- Hacks, C. (2012). Extreme range links: Lora 868 / 900mhz sx1272 lora module for arduino waspmote and raspberry pi. url <https://www.cooking-hacks.com/documentation/tutorials/extreme-range-lora-sx1272-module-shield-arduino-raspberry-pi-intel-galileo>.
- Krupka, L., Vojtech, L., and Neruda, M. (2016). The issue of lpwan technology coexistence in iot environment. , pages 1–8.
- Lei, K., Ma, Y., and Tan, Z. (2014). Performance comparison and evaluation of web development technologies in php, python, and node. js. In *2014 IEEE 17th international conference on computational science and engineering*, pages 661–668. IEEE.

- Li, L., Ren, J., and Zhu, Q. (2017). On the application of lora lpwan technology in sailing monitoring system. In *2017 13th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*, pages 77–80. IEEE.
- Lora-Alliance (2018). what is LoRaWAN?, tomado de <https://www.lora-alliance.org/what-is-lora>.
- Lora-Alliance (2019). Lora alliance®. url [www.lora-alliance.org/](http://www.lora-alliance.org/).
- mbed (2019). Sx1272mb2xas. url <https://os.mbed.com/components/SX1272MB2xAS/>.
- Mekki, K., Bajic, E., Chaxel, F., and Meyer, F. (2019). A comparative study of lpwan technologies for large-scale iot deployment. *ICT express*, 5(1):1–7.
- ModMyPi LTD (2019). Raspberry Pi Comparison Table tomado de <https://os.mbed.com/components/SX1272MB2xAS/>.
- Montenbruck, O., Schmid, R., Mercier, F., Steigenberger, P., Noll, C., Fatkuln, R., Kogure, S., and Ganeshan, A. (2015). Gnss satellite geometry and attitude models. *Advances in Space Research*, 56(6):1015–1029.
- Ordoñez, B., Yamiith, C., et al. (2017). Implementación de una red de sensores inalámbricos lpwan mediante módulos lora para el monitoreo de la calidad del agua en 2 ríos. .
- 7  
Rademakers, E., De Bakker, P., Tiberius, C., Janssen, K., Kleihorst, R., and El Ghouti, N. (2016). Obtaining real-time sub-meter accuracy using a low cost gnss device. In *2016 European Navigation Conference (ENC)*, pages 1–8. IEEE.
- Raspberry (2018). Raspberry pi zero w. url <https://www.raspberrypi.org/products/raspberry-pi-zero-w/>.
- Sabina, J. Á. R., Valle, D. G., Ruiz, C. P., García, J. M. M., and Laguna, A. G. (2015). Aerial photogrammetry by drone in archaeological sites with large structures. methodological approach and practical application in the medieval castles of campo de montiel. *Virtual Archaeology Review*, 6(13):5–19.
- Semtech, A. (2015). Lora™ modulation basics, application note.
- SigFox (2019). SigFox,Tomado de <https://www.sigfox.com/en/>.
- Simón Toquero, B. et al. (2011). Levantamiento mediante gnss de una red de puntos anclazando con precisión los distintos espacios de la upct en el mismo sistema de coordenadas. .
- Sinha, R. S., Wei, Y., and Hwang, S.-H. (2017). A survey on lpwa technology: Lora and nb-iot. *Ict Express*, 3(1):14–21.
- u blox, D. (2019). ZED-F9P u-blox F9 high precision GNSS moduleData Sheet. In *u-blox*.

Urbano-Molano, F. A. (2013). Redes de sensores inalámbricos aplicadas a optimización en agricultura de precisión para cultivos de café en colombia. *Journal de Ciencia e Ingeniería*, 5(1):46–52.

wikipedia (2019). Topología de red,Tomado de <https://es.wikipedia.org/wiki/Topolog>



# A

## APÉNDICES

### A.1. CONFIGURACIÓN DE RASPBIAN

- Ingresar a <https://www.raspberrypi.org/downloads/raspbian/> y descargar la versión lite.
- Descargar SD Card Formatter - <https://www.sdcard.org/downloads/formatter/>
- Descargar balenaEtcher <https://www.balena.io/etcher/>
- Ingresa la tarjeta microSD al computador y abrir SD Card Formatter ,seleccionar la unidad donde se inserto la SD y dar "format" como se muestra en la figura A.1.
- Abrir Etcher y cargar la imagen iso descargada de Raspbian y darle "Flash" como se muestra en la figura A.2.
- Retirar la tarjeta SD y re insertar en el computador.
- Crear un archivo sin extensión llamado "ssh".
- Crear un archivo llamado wpa\_supplicant.conf, dentro del archivo agregar la siguiente datos :

```
1 ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
2 update_config=1
3 country=CO
4 network={
5     ssid="Nombre De La Red"
6     psk="pass"
7     key_mgmt=WPA-PSK
8 }
```

- Agregar los dos archivos creados a la SD en la carpeta Raíz.
- Poner la tarjeta SD en la raspberry y conectar a la corriente.

A

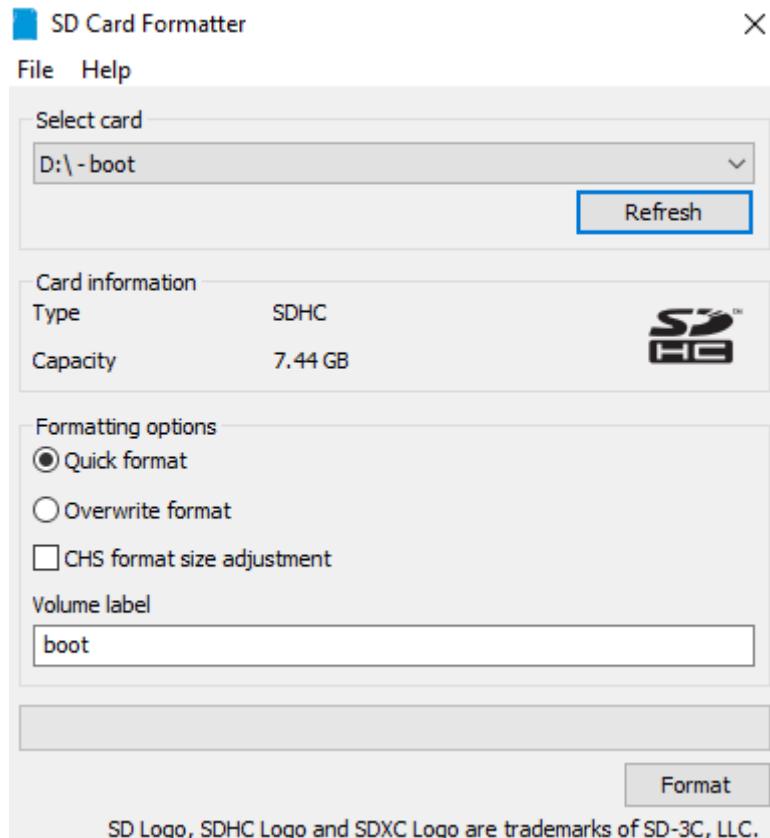


Figura A.1: Interfaz de SD Card Formatter  
Fuente: Propia

- Desde nuestro computador genera una red WI-Fi con el "Nombre De La Red" establecidos en el archivo de configuración `wpa_supplicant.conf`, la raspberry se conectara automáticamente a la red establecida obteniendo un dirección IP.
- Obtenemos la IP de la raspberry ya sea mirando la configuración del computador o por medio de un programa como lo es "Advanced IP Scanner".
- Con la IP de la Raspberry en la terminal de PuTTY <https://www.putty.org/> ingresamos la IP como se muestra en la figura A.3
- Para iniciar sección en la raspberry usamos el usuario "pi" la contraseña raspberry".
- Con el comando "sudo raspi-config" nos muestra opciones como se ilustra en la figura A.4, donde podemos cambiar la contraseña así como tambien en la opciones de "Interfacing Options" habilitamos SPI, I2C y Serial.

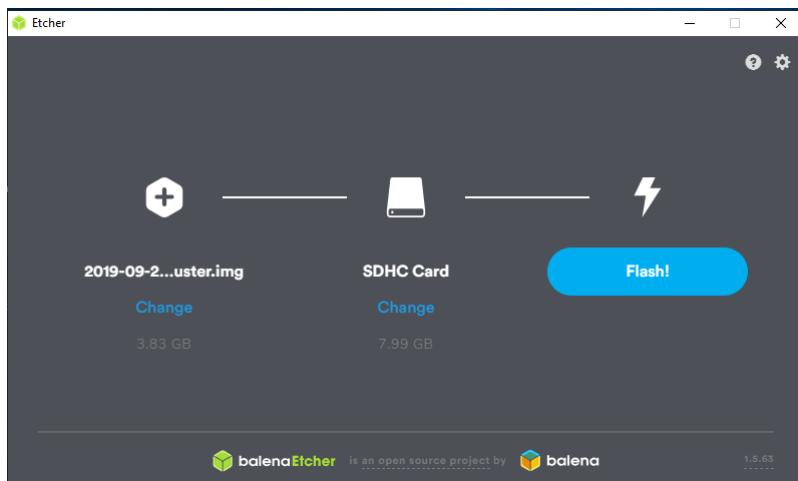


Figura A.2: Interfaz Etcher

Fuente: Propia

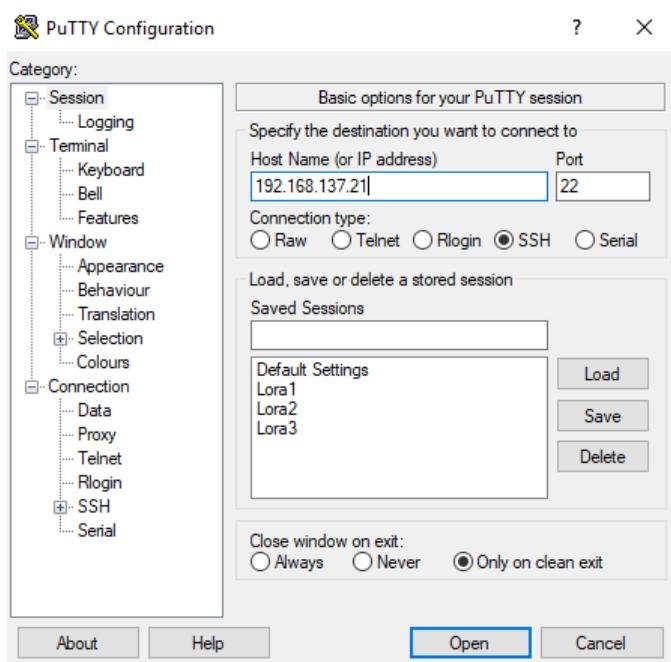


Figura A.3: Interfaz de PuTTy

Fuente: Propia

## A

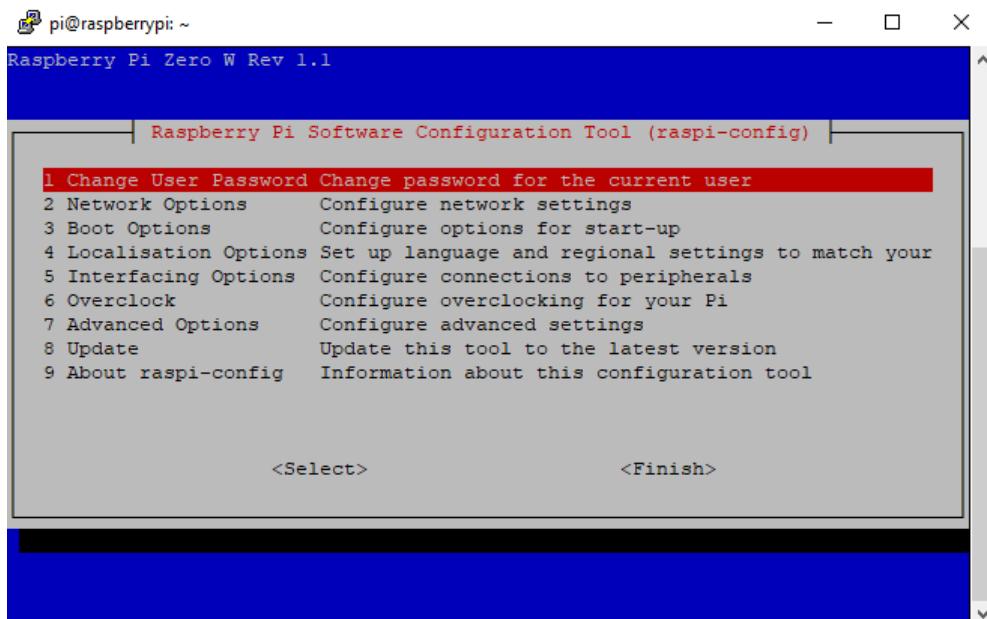


Figura A.4: Raspi-Config  
Fuente: Propia

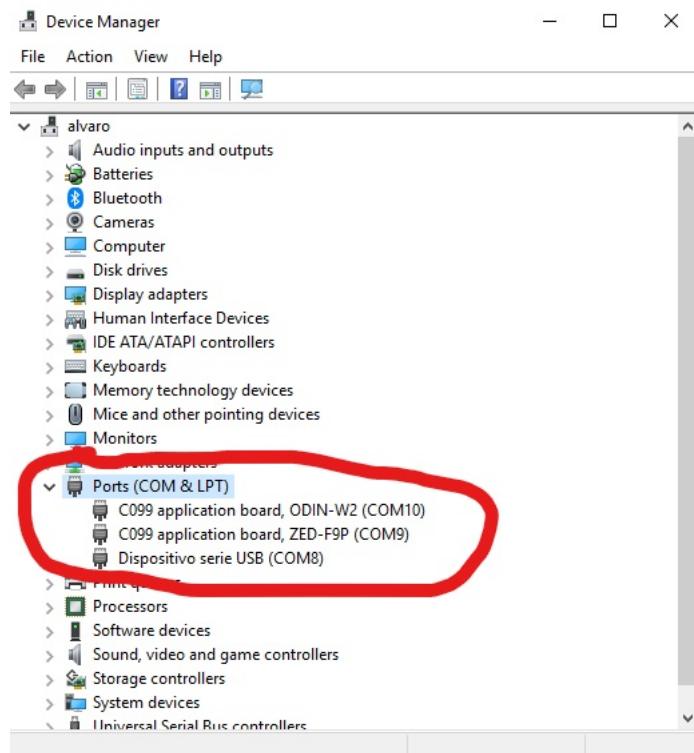


Figura A.5: Administrador de Dispositivos

Fuente: Propia

## A.2. CONFIGURACIÓN DE U-BLOX ZED-F9P

Para configurar el moduló ublox F9P para recibir datos RAW se realizaron los siguientes pasos:

- Conectar el modulo F9P a un computador de mesa o portátil por medio del cable usb así como tambien a una fuente de poder.
- En Administradora de dispositivos de windows como se muestra en la figura A.5 se observa que aparecen 3 dispositivos,el que configuraremos sera el que se encuentra en el puerto 8.
- Descargamos e instalamos u-center de ublox <https://www.u-blox.com/en/product/u-center/>
- en u-center nos conectamos al puerto 8
- al conectarlos el puerto 8 vemos la información que actualmente recibe nuestro modulo como en la parte superior derecha en la ventana "View" desplegamos las ventanas Messages View , Configuration View y Generation 9 Configuration view.

A

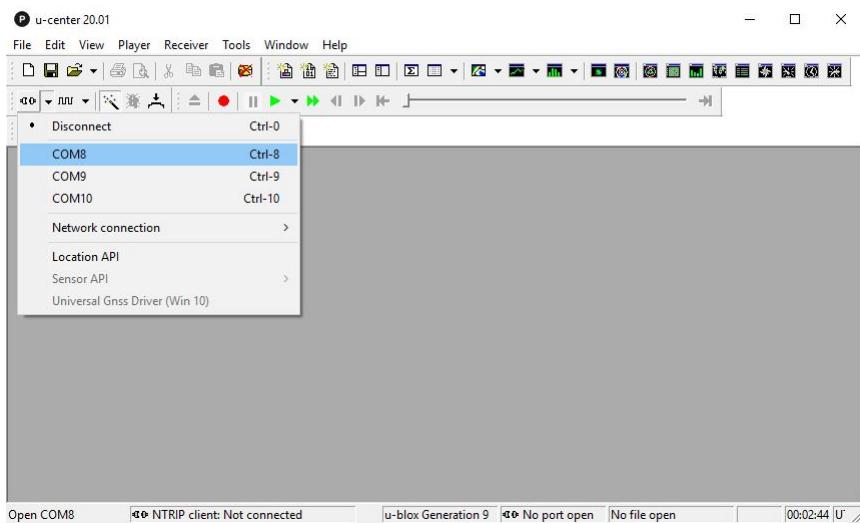
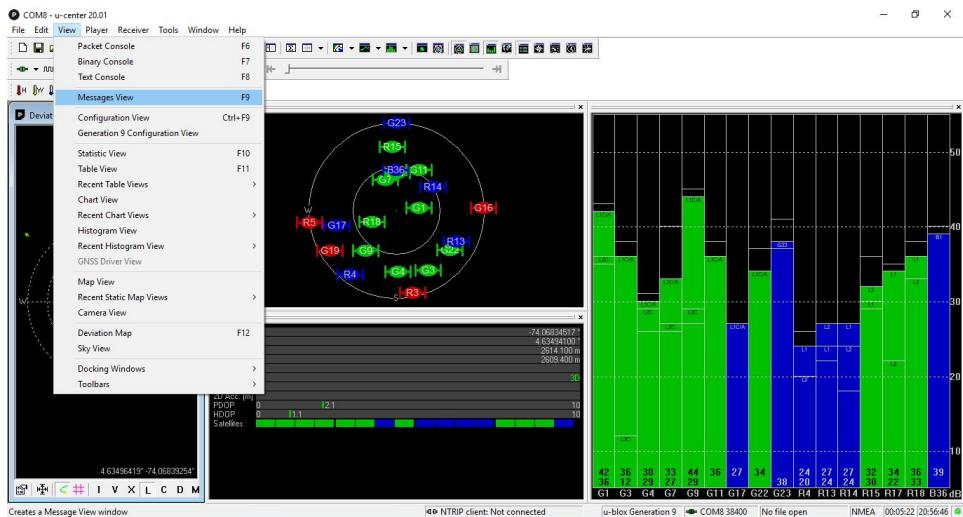


Figura A.6: Seleccionar puerto U-Center

Fuente: Propia

Figura A.7: U-Center  
Fuente: Propria

- En la ventana Generation 9 Configuration view A.8 revisamos que todos los satélites estén seleccionados ,así como la flash y RAM y le damos en "Send Configuration"

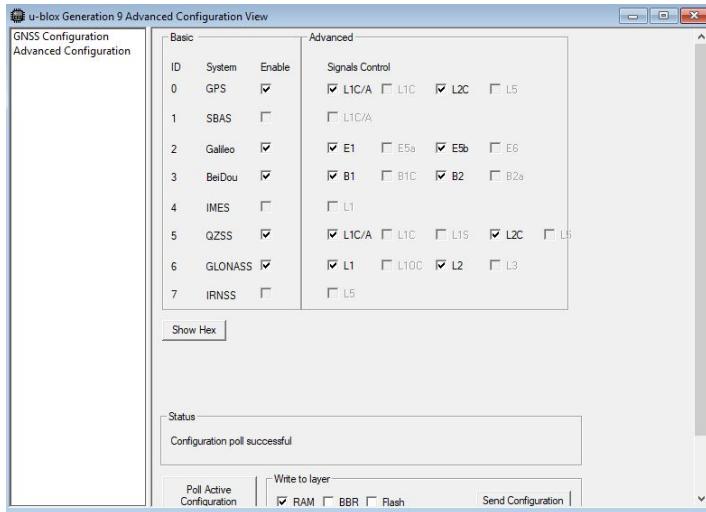


Figura A.8: Generation 9 Configuration view

Fuente: Propia

- En la ventana de message view A.9 deshabilitamos la que dice NMEA dandole click derecho y disabled child Message , y habilitando en UBX en la sección RXM los de nombre RAWX y SFRBX.
- En la ventana de Configure View A.10 el ultimo de la lista "VALSET" seleccionamos el grupo que dice CFG-MSGOUT y cargamos a la lista las llaves RTCM 1074, 1077, 1084, 1087, 1094, 1097, 1124, 1127 y cambiamos el valor de todas a 1.
- En la misma ventana en PRT(Ports) establecemos la configuración que deseamos para la comunicacion
- En la misma ventana en RATE(Rates) si es para un GNSS que va a servir de base tomamos el periodo de medicion en 200ms si es para orto fin se elije el tiempo de medicion en 1000ms.
- Para finalizar en esta ventana el CFG(Configuration) elejimos Save currente Configuration y los cuatro dispositivos, posteriormente realizamos en envío de la configuración con el botón "SEND" ubicado en la parte inferior izquierda.

A

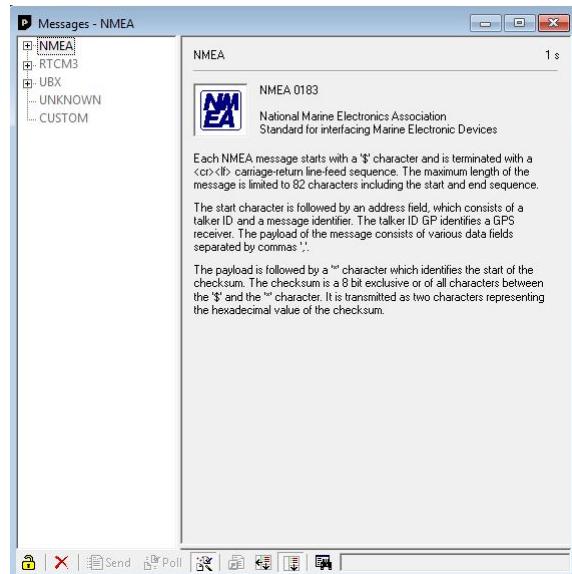


Figura A.9: Message View  
Fuente: Propia

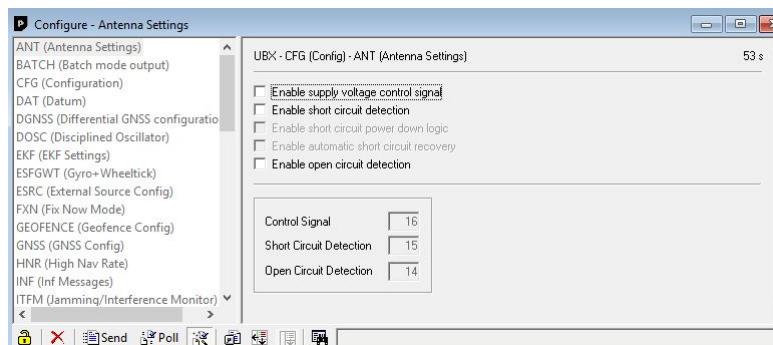


Figura A.10: Configure View  
Fuente: Propia

## A.3. MANUAL USUARIO

En esta primera vista se tiene un inicio de sesión para acceder al aplicativo en django, para configurar un nuevo usuario es necesario hacerlo por medio del comando **python manage.py createsuperuser** donde ingresaremos el usuario y nombre que queramos, Posteriormente podremos acceder por medio de este en la figura A.11 donde ponemos el nombre y la contraseña.

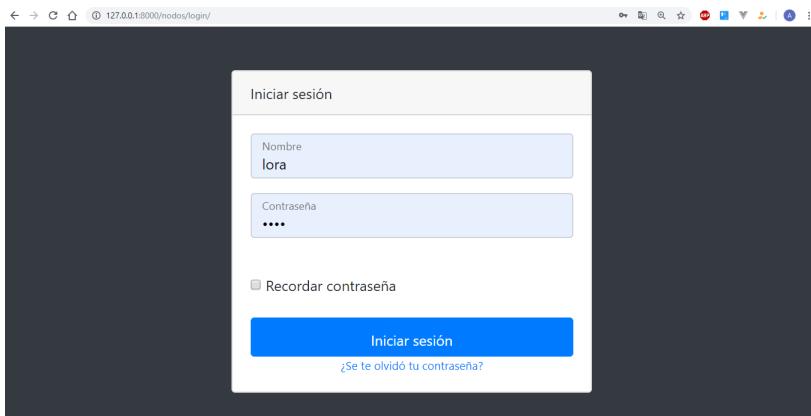


Figura A.11: Detalles de Nodo Django  
Fuente: Propia

Al iniciar sesión se observara una vista como en la figura A.12 , donde se ubica una lista con los nodos que han esta conectados a este nodo así como un mapa de la ultima ubicación de los nodos, tambien hay botón actualizar que permite a django recoger los datos que se están generando en la red,tambien se puede acceder a la información especifica de un nodo al hacer click sobre el nodo que queramos.

En esta imagen A.13 se muestran los datos que se han recopilado respecto a un nodo ya sea tanto en posición (latitud,longitud,altitud,etc) como la información pertinente al funcionamiento de nodo (Paquetes enviados,paquetes recibidos,etc).

En la parte inferior se ubica un botón para la descarga de los datos que ha recibido el nodo respecto a este nodo, así como un botón para poder visualizarlos desde la interfaz gráfica, y por ultimo un botón para regresar a la lista de nodos disponibles.

A

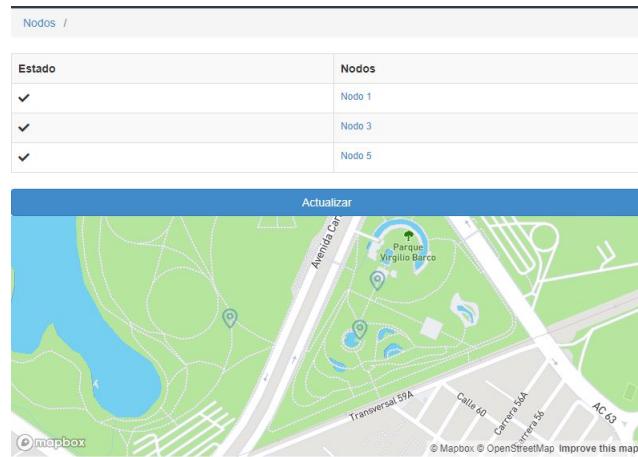


Figura A.12: Detalles de Nodo Django

Fuente: Propia

Nodo 5	
LoRa	
Nombre	Estado
Estado	✓
Numero de Nodo	5
Ancho de Banda	0
Correinte	240
Paquetes Enviados	40
Paquetes Recibidos	40
Tiempo Envío	0.0
RSSI(Fuerza Señal)	-104
Payload (Carga Util)	255

GNSS	
Nombre	Estado
Estado	✓
Numero Satélites	12
Dilución Horizontal de la Posición	0.65
Latitud	4.6348666666666665
Longitud	-74.06821616666667
Altitud	2588.9
Fix Quality	1

Figura A.13: Detalles de Nodo Django

Fuente: Propia

## A.4. CÓDIGO CONFIGURACIÓN DE NODOS

A

```
1 #include "stdio.h"
2 #include "time.h"
3 #include "arduPiLoRa.h"
4 int e, paqueteEnviado = 0, paqueteRecibido = 0, paqOrigin = 0;
5 int snr, rssi, rssil, proxNodo;
6 int posicion = 1;
7 int p = 0;
8 int valor = 0;
9 char my_packet[300];
10 char mess[] = "echo ''";
11 char mesEnd[] = ".txt";
12 bool mode = false;
13 bool mode1 = true;
14 double secs = 0;
15 time_t rawtime;
16 char info1[300];
17 char info2[300];
18 char infoT[300];
19 char toGnss[] = "sudo python //home//pi//Desktop//LoRa//gnss.py ";
20 char toSave[] = "sudo python //home//pi//Desktop//LoRa//saveData.py ";
21 char coma[] = " ";
22 char mgsA[] = "a";
23 char mgsB[] = "b";
24 char aux[1];
25 int numNodo =1;
26 char buff[255];
27 char stringNodo[] = "//home//pi//Desktop//data//nodo";
28 void setup(){
29     // Power ON the module
30     e = sx1272.ON();
31     printf("Setting power ON: state %d\n", e);
32
33     // Set transmission mode
34     e |= sx1272.setMode(4);
35     printf("Setting Mode: state %d\n", e);
36
37     // Set header
38     e |= sx1272.setHeaderON();
39     printf("Setting Header ON: state %d\n", e);
40
41     // Select frequency channel
42     e |= sx1272.setChannel(CH_10_868);
43     printf("Setting Channel: state %d\n", e);
44
45     // Set CRC
46     e |= sx1272.setCRC_ON();
47     printf("Setting CRC ON: state %d\n", e);
48
49     // Select output power (Max, High or Low)
50     e |= sx1272.setPower('H');
51     printf("Setting Power: state %d\n", e);
52
53     // Set the node address
54     e |= sx1272.setNodeAddress(numNodo);
55     printf("Setting Node address: state %d\n", e);
```

## A

```

56 // Print a success message
57 if (e == 0)
58     printf("SX1272 successfully configured\n");
59 else
60     printf("SX1272 initialization failed\n");
61 delay(1000);
62 }
63 void esclavo(void){
64     e = sx1272.receivePacketTimeoutACK(10000);
65     if (e == 0) {
66         for (unsigned int i = 0; i < sx1272.packet_received.length; i++)
67         {
68             my_packet[i] = (char)sx1272.packet_received.data[i];
69         }
70         if (my_packet[0] == 97) {
71             paqOrigin = (int)sx1272.packet_received.src;
72             paqueteRecibido = paqueteRecibido + 1;
73             e = 1;
74             int error = 0;
75             while (e > 0){
76                 struct timeval start, stop;
77                 gettimeofday(&start, NULL);
78                 if (mode1){
79                     e = sx1272.sendPacketTimeoutACK(paqOrigin, buff);
80                 }else{
81                     e = sx1272.sendPacketTimeoutACK(paqOrigin, mgsB);
82                     if(e == 0){
83                         mode = true;
84                     }
85                 }
86                 if (e == 0){
87                     gettimeofday(&stop, NULL);
88                     secs = (double)(stop.tv_usec - start.tv_usec) / 1000000 + (
89                     double)(stop.tv_sec - start.tv_sec);
90                     paqueteEnviado = paqueteEnviado + 1;
91                     printf("Info regresada %s:\n", buff);
92                 }
93                 if (error==2){
94                     break;
95                 }
96                 error = error +1;
97             }
98         }
99     }
100 void maestro(void){
101     e = sx1272.getNodeAddress();
102     int k = sx1272._nodeAddress;
103     for (int i = 1; i < 6; i++) {
104         if (i != k) {
105             printf("Pregunta al Nodo %d\n", i);
106             e = sx1272.sendPacketTimeoutACK(i, mgsA);
107             if (e == 0) {
108                 printf("Enviada informacion a nodo %i\n", i);
109                 e = sx1272.receivePacketTimeoutACK(10000);
110                 if (e == 0){

```

```
111     for (unsigned int j = 0; j < sx1272.packet_received.length; j
112     ++){  
113         my_packet[j] = (char)sx1272.packet_received.data[j];}  
114         printf("Info regresada %s\n", my_packet);  
115         if (strcmp (mgsB, my_packet) == 0){  
116             mode = false;  
117         }else{  
118             sprintf(info1, "%s%d%s%s", toSave,i, coma ,my_packet);  
119             system(info1);  
120         }  
121     }  
122 }  
123 }  
124 }  
125  
126 int toString(char a[]){  
127     int c, sign, offset, n;  
128  
129     if (a[0] == '-'){ // Handle negative integers  
130         sign = -1;  
131     }  
132     if (sign == -1){ // Set starting position to convert  
133         offset = 1;  
134     }  
135     else {  
136         offset = 0;  
137     }  
138     n = 0;  
139     for (c = offset; a[c] != '\0'; c++) {  
140         n = n * 10 + a[c] - '0';  
141     }  
142     if (sign == -1) {  
143         n = -n;  
144     }  
145     return n;  
146 }  
147  
148 void createInfo(void){  
149     e = sx1272.getNodeAddress();  
150     e = sx1272.getRSSI();  
151     e = sx1272.getBW();  
152     e = sx1272.getMaxCurrent();  
153     e = sx1272.getPayloadLength();  
154     sprintf(info1, "%s%d%s%d%s%d%s%d%s%f%s%d%s%d", toGnss,sx1272.  
_nodeAddress, coma, sx1272._bandwidth, coma, sx1272._maxCurrent,  
coma, paqueteEnviado, coma, paqueteRecibido, coma, secs, coma,  
sx1272._RSSI, coma, sx1272._payloadlength);  
155     int a =system(info1);  
156     if (a == 0){  
157         FILE *fp;  
158         fp = fopen(infoT, "r");  
159         fgets(buff, 255, (FILE*)fp);  
160         fclose(fp);  
161     }  
162     int d =system("sudo python //home//pi//Desktop//LoRa//modo.py");  
163     if (d == 0){
```

**A**

```
164     mode1=true;
165 }else{
166     mode1=false;
167 }
168 }
169 void datos(){
170     e = sx1272.getNodeAddress();
171     sprintf(infoT, "%s%d%s", stringNodo,sx1272._nodeAddress,mesEnd);
172     FILE *fp;
173     fp = fopen(infoT, "r");
174     fgets(buff, 255, (FILE*)fp);
175     fclose(fp);
176     char *p;
177     int j = 1;
178     p = strtok (buff, ",");
179     while (p!= NULL){
180         printf("Info Leida %d\n",toString(p));
181         if (j == 4){paqueteEnviado = toString(p);}
182         if (j == 5){paqueteRecibido = toString(p);}
183         p = strtok (NULL, ",");
184         j = j+1;
185     }
186 }
187
188 int main(){
189     setup();
190     datos();
191     while (1){
192         if (mode){maestro();}
193         else{esclavo();}
194         createInfo();
195     }
196     return (0);
197 }
198 }
```

## A.5. CÓDIGO LECTURA DE GNSS

```

1 import serial
2 import time, os
3 import pynmea2
4 import sys
5 import os
6 from datetime import datetime
7
8 nodo=str(sys.argv[1])
9 bw=str(sys.argv[2])
10 maxCurrent=str(sys.argv[3])
11 paqueteEnviado=str(sys.argv[4])
12 paqueteRecibido=str(sys.argv[5])
13 tiempoEnvio =str(sys.argv[6])
14 RSSI=str(sys.argv[7])
15 payloadlength=str(sys.argv[8])
16 i = 0
17 infoLora = str(nodo) + ',' + str(bw) + ',' + str(maxCurrent)+ ',' + str(
    paqueteEnviado) + ',' + str(paqueteRecibido) + ',' + str(tiempoEnvio)
    + ',' + str(RSSI) + ',' + str(payloadlength)
18 data = ''
19 # nodo.txt se refiere a la informacion propia de nodo tomada desde el
    sensor
20 # Se deja un tiempo para que el GNSS inicie y de informacion util de
    ubicacion
21 fi=open("/home/pi/Desktop/data/nodo"+ str(nodo)+".txt", "r")
22 conteente =fi.read()
23 def line_prepender(nodo, line):
24     with open("/home/pi/Desktop/data/nodo"+ str(nodo)+".txt", "w") as
        f:
25         f.seek(0,0)
26         f.write(str(line)+'\n'+conteente)
27         f.close()
28 now = datetime.now()
29
30 try:
31     port = serial.Serial(port="/dev/ttyACM0",
32                           baudrate=9600,
33                           timeout=0.1,
34                           bytesize=serial.EIGHTBITS)
35     port.close()
36     port.open()
37     while True:
38         data = port.readline()
39         if (data.startswith("$GNGGA")):
40             nmea = str(infoLora) + '_' + str(data.rstrip('$\r\n')) + '_'
41             msg = pynmea2.parse(data)
42             if msg.gps_qual == 1:
43                 i = 1
44                 line_prepender(nodo,nmea)
45             if i == 1:
46                 break
47             now2 = datetime.now()
48             asd = now2 -now
49             if asd.seconds > 60:
50                 break

```

**A**

```
51 except :  
52     nmea =nodo+",0,0,0,0,0,0,$"  
53     line_prepender(nodo,nmea)  
54  
55 }
```

## A.6. REPOSITORIO GITHUB

Toda la información se puede encontrar en el siguiente repositorio :  
<https://github.com/AndresCUd/tesis>