

# ANÁLISIS DE TECNOLOGÍAS RADIO PARA LA TRANSMISIÓN DE INFORMACIÓN AL USUARIO DE UN SISTEMA DE CONTROL DE FLOTA



**María Alejandra Palacios Ibarra**  
**Stephany Lucía Zúñiga Pérez**

*Universidad del Cauca*

**Facultad de Ingeniería Electrónica y Telecomunicaciones**  
**Departamento de Telecomunicaciones**  
**GNTT Grupo de Nuevas Tecnologías en Telecomunicaciones**  
**GRIAL Grupo de Radio e Inalámbricas**  
**Popayán, 2019**

# ANÁLISIS DE TECNOLOGÍAS RADIO PARA LA TRANSMISIÓN DE INFORMACIÓN AL USUARIO DE UN SISTEMA DE CONTROL DE FLOTA



Trabajo de grado presentado como requisito para obtener el título de Ingeniero en Electrónica y Telecomunicaciones

**María Alejandra Palacios Ibarra**  
**Stefhany Lucía Zúñiga Pérez**

Directora: Msc. Claudia Milena Hernández Bonilla

*Universidad del Cauca*

**Facultad de Ingeniería Electrónica y Telecomunicaciones**  
**Departamento de Telecomunicaciones**  
**GNTT Grupo de Nuevas Tecnologías en Telecomunicaciones**  
**GRIAL Grupo de Radio e Inalámbricas**  
**Popayán, 2019**

## Tabla de contenido

1. CAPÍTULO 1: INTRODUCCIÓN .....	1
1.1 CIUDADES INTELIGENTES.....	2
1.1.1. Desarrollo de una Ciudad Inteligente.....	3
1.1.2. Estandarización para una Ciudad Inteligente .....	4
1.2. SISTEMAS DE TRANSPORTE INTELIGENTE (ITS) .....	4
1.2.1. Sistema de Transporte Público Inteligente (PITS).....	6
1.3. SISTEMA DE CONTROL DE FLOTA.....	7
1.3.1. Información al usuario .....	8
2. CAPÍTULO 2: ANÁLISIS DE TECNOLOGÍAS RADIO PARA LA TRANSMISIÓN DE INFORMACIÓN EN UN SISTEMA DE CONTROL DE FLOTA	9
2.1. PARÁMETROS DE ANÁLISIS .....	9
2.2. ANÁLISIS DE TECNOLOGÍAS RADIO.....	9
2.2.1. Tecnologías LPWAN .....	13
3. TECNOLOGÍA LoRa.....	23
3.1. LoRa ALLIANCE .....	23
3.2. LoRa Y LoRaWAN .....	23
3.2.1. LoRa.....	23
3.3. CLASES DE DISPOSITIVOS LoRaWAN .....	27
3.4. ARQUITECTURA DE LA RED LoRaWAN .....	28
3.5. SEGURIDAD LoRaWAN .....	28
3.6. TRAMA DE COMUNICACIONES.....	29
3.7. REGULACIÓN LoRa.....	30
3.7.1. LoRa para Europa .....	30
3.7.2. LoRa para América Del Norte.....	30
4. CAPITULO 4: METODOLOGÍA DE DESARROLLO .....	32
4.1. REQUERIMIENTOS.....	33
4.1.1. Requisitos funcionales.....	33
4.1.2. Requisitos no funcionales.....	33
4.1.3. Restricciones del sistema .....	33

4.2.	ANÁLISIS Y DISEÑO .....	34
4.2.1.	Tecnología radio para la transmisión de información en el piloto .....	34
4.2.2.	Diseño del piloto de información de control de flota .....	34
4.3.	IMPLEMENTACIÓN .....	58
4.3.1.	Implementación de la comunicación LoRa y conexiones del MG .....	58
4.3.2.	Implementación Software .....	62
4.3.3.	Implementación hardware .....	66
4.4.	PRUEBAS Y ANÁLISIS DE RESULTADOS .....	71
4.4.1.	Elección del escenario .....	71
4.4.2.	Configuración del escenario .....	74
4.4.3.	Prueba de confiabilidad del piloto .....	77
4.4.4.	Prueba de cobertura del piloto .....	86
5.	CAPITULO 5: CONCLUSIONES Y TRABAJOS FUTUROS .....	91
5.1.	CONCLUSIONES .....	91
5.2.	TRABAJOS FUTUROS .....	92
	Bibliografía .....	94

## Lista de Figuras

Figura 1.1 Componentes de un Sistema de Transporte Inteligente (Por los autores).....	5
Figura 1.2 Componentes del PITS (Por los autores).....	6
Figura 3.1 Influencia de la Tasa de Codificación en la sensibilidad del receptor [54]. ....	26
Figura 3.2 Pila de protocolos para la tecnología LoRa [41]. ....	27
Figura 3.3 Arquitectura de la Red LoRaWAN [41]. ....	28
Figura 3.4 Trama de comunicación LoRa [54]. ....	30
Figura 4.1 Diagrama en bloques de la metodología RUP. Por los autores. ....	32
Figura 4.2 Modelo esquemático del piloto .....	35
Figura 4.3 Diagrama de flujo para establecer una comunicación LoRa entre dispositivos radio. ....	37
Figura 4.4 Modelo esquemático del MaB.....	38
Figura 4.5 Diseño esquemático del MaB. Por los autores.....	39
Figura 4.6 Disponibilidad de pines LoRa/GPS Shield v9. Por los autores. ....	40
Figura 4.7 Diseño circuital del GPS y Arduino uno. Por los autores.....	42
Figura 4.8 Diagrama circuital LoRa/GPS Shield montado sobre Arduino con el módulo RFID RC-522. Por los autores. ....	42
Figura 4.9 Diagrama de flujo para el MaB. Por los autores. ....	44
Figura 4.10 Modelo esquemático del MG .....	45
Figura 4.11 Diagrama de flujo para el MG. Por los autores.....	47
Figura 4.12 Modelo esquemático del MP.....	48
Figura 4.13 Modelo esquemático del MP. Por los autores. ....	49
Figura 4.14 Disponibilidad de pines LoRa Shield v95 .Por los autores. ....	49
Figura 4.15 Diagrama circuital LoRa Shield v95 montado sobre Arduino Mega 2560 y la LCD 1602. Por los autores. ....	50
Figura 4. 16 Diagrama de flujo general para el MP. Por los autores. ....	52
Figura 4.17 Modelo esquemático del MS.....	53
Figura 4.18 Diseño esquemático de la comunicación entre servidores y la página Web.....	54
Figura 4.19 Diagrama de flujo MS. Por los autores. ....	55
Figura 4.20 Diseño general del piloto de información de control de flota. Por los autores.....	56
Figura 4.21 Diagrama de flujo del piloto de información al usuario. Por los autores.....	57
Figura 4. 22 Pseudocódigo para establecer comunicación LoRa entre dispositivos radio.....	59
Figura 4. 23 Pseudocódigo de la transmisión de paquetes en LoRa. ....	60
Figura 4.24 Pseudocódigo de la recepción de paquetes en LoRa .....	60
Figura 4.25 Conexión Gateway LoRa OLG01. Por los autores. ....	61
Figura 4.26 Pseudocódigo para la obtención del número de ruta .....	62
Figura 4. 27 Pseudocódigo para la obtención de ubicación del bus en coordenadas GPS .....	63
Figura 4. 28 Pseudocódigo proceso en el MG .....	64
Figura 4. 29 Pseudocódigo del proceso realizado en MP .....	65
Figura 4. 30 Pseudocódigo para el desarrollo de la página Web .....	66
Figura 4.31 Conexión LoRa/GPS Shield V95 con el Arduino uno. ....	67
Figura 4.32 Conexión de transmisión y recepción de datos entre GPS y Arduino uno. ....	67
Figura 4.33 Implementación hardware MaB. ....	68
Figura 4. 34 Dispositivos LoRa Shield V95 y Arduino UNO acoplados .....	69
Figura 4.35 Implementación hardware MP culminada.....	69
Figura 4.36 Presentación de la información al usuario mediante LCD.....	70
Figura 4.37 Visualización en mapa de la página Web.....	70
Figura 4. 38 Visualización satelital en la página Web.....	71

Figura 4.39 Avenida Pomona (escenario de pruebas).	72
Figura 4.40. Ubicación de las paradas.	73
Figura 4.41 Diagrama esquemático del escenario.	73
Figura 4.42 Montaje para obtener el patrón de radiación de la antena del MG.	75
Figura 4.43 Patrón de radiación de la antena del MG con paso angular $=2^\circ$ .	76
Figura 4.44 Patrón de radiación de la antena del MG con paso angular $=0.5^\circ$ .	76
Figura 4.45 Prueba del margen de error del GPS LoRa.	79
Figura 4.46 Tiempo de espera del piloto en P1.	81
Figura 4.47 Tiempo de espera medido por cronometro en P1.	81
Figura 4.48 Tiempo de espera del piloto en P2.	82
Figura 4.49 Tiempo de espera medido por cronometro en P2.	82
Figura 4.50 Tiempo de espera del piloto en P3.	83
Figura 4.51 Tiempo de espera medido por cronometro en P3.	83
Figura 4.52 Cobertura del MG con el método Rec. UIT-R P.526-13.	88
Figura 4.53 Cobertura del MG con el método Cost 231.	88
Figura 4.54 Mapa de cobertura del MG basado en pruebas experimentales.	89
Figura A.1 Cilindro transverso.	100
Figura A.2 Cilindro transverso 2.	101
Figura B.1 Red Dragino-XXX generada por la Gateway OLG01.	103
Figura B.2 Paso a paso para llegar a la casilla preferencias en la interfaz de Arduino.	104
Figura B.3 Interfaz Preferencias en Arduino.	105
Figura B.4 Selección Gestor de Tarjetas en la interfaz de Arduino.	106
Figura B.5 Selección e instalación de la tarjeta Dragino Yun.	107
Figura C.1 Interfaz servidor ThingSpeak para el registro de usuarios.	108
Figura C.2 Interfaz de ThingSpeak para la creación de un canal.	109
Figura C.3 Selección de la pestaña API Keys en la interfaz de ThingSpeak.	110
Figura C.4 Claves proporcionadas por el servidor ThingSpeak.	110
Figura D.1 Creación de proyecto en Google Cloud Platform.	111
Figura D.2 Paso a paso para llegar a la pestaña de credenciales.	112
Figura D.3 Paso a paso para la creación de claves APIs.	112
Figura D.4 Clave API Key ya creada.	113

## Lista de tablas

Tabla 2.1 Análisis grupos de tecnologías radio .....	13
Tabla 2.3 Características técnicas de tecnologías LPWAN (1) [28] [30]. .....	19
Tabla 2.4 Características técnicas de tecnologías LPWAN (2) [28][30] .....	20
Tabla 2.5 Comparativo entre tecnologías LPWAN (1) .....	21
Tabla 2.6 Comparativo entre tecnologías LPWAN (2) .....	21
Tabla 3.1 Influencia del BW en el tiempo en aire y sensibilidad [54]. .....	24
Tabla 3.2 Influencia del SF en el tiempo en aire y sensibilidad [54]. .....	25
Tabla 3.3 Influencia de la Tasa de Codificación en el tiempo en aire [54]. .....	25
Tabla 4.1 Configuraciones para dispositivos radio LoRa [73]. .....	36
Tabla 4.2 Disponibilidad de pines LoRa/GPS Shield v95. Por los autores. ....	41
Tabla 4.3 Asignación de pines entre Arduino uno y módulo RFID RC-522. Por los autores. ....	43
Tabla 4.4 Asignación de pines entre Arduino MEGA, LCD y Potenciómetro. Por los autores. ....	51
Tabla 4.5 Coordenadas geográficas de las paradas. ....	74
Tabla 4.6 Valor de tiempo de espera teórico de cada parada .....	78
Tabla 4.7 Comparación de coordenadas del GPS de referencia y el GPS LoRa .....	80
Tabla 4.9 Comparativo entre $t_t$ , $t_c$ y $t_{vp}$ .....	<b>¡Error! Marcador no definido.</b>
Tabla 4.10 Parámetros de configuración Xirio [9]. .....	87

## Lista de Acrónimos

<b>3GPP</b>	<i>3rd Generation Partnership Project</i> , Proyecto de Asociación de Tercera Generación.
<b>ADR</b>	<i>Adaptive Data Rate</i> , Velocidad de Datos Variable.
<b>AES</b>	<i>Advanced Encryption Algorithm</i> , Estándar de Cifrado Avanzado.
<b>AES-CMAC</b>	<i>AES-Cipher-Based Message Authentication Code</i> , AES-Código de Autenticación de Mensaje Basado en Cifrado.
<b>BCC</b>	<i>Binary Convolutional Codes</i> , Códigos Convolucionales Binarios
<b>BPSK</b>	<i>Phase Shift Keying Binary</i> , Modulación por Desplazamiento de Fase Binario.
<b>BS</b>	<i>Base Station</i> , Estación Base.
<b>CDMA</b>	<i>Code Division Multiple Access</i> , Acceso Múltiple por División de Código
<b>CR</b>	<i>Code Rate</i> , Tasa de Codificación.
<b>CRC</b>	<i>Cyclic Redundancy Check</i> , Verificación de Redundancia Cíclica.
<b>CSS</b>	<i>Chirp Spread Spectrum</i> , Modulación de Espectro Ensanchado de Chirp.
<b>CTR</b>	<i>Counter Mode Encryption</i> , Cifrado de Modo Contador.
<b>DBPSK</b>	<i>Differential Binary Phase Shift Keying</i> , Desplazamiento de Fase Diferencial Binaria.
<b>EC-GSM-IoT</b>	<i>Extended Coverage-GSM-IoT</i> , Cobertura Ampliada GSM-IoT.
<b>EDACS</b>	<i>Enhanced Digital Access Communications System</i> , Sistema de Comunicaciones de Acceso Digital Mejorado.
<b>ETSI</b>	<i>European Telecommunications Standards Institute</i> , Instituto Europeo de Estandarización de las Telecomunicaciones.
<b>FCC</b>	<i>Federal Communications Commission</i> , Comisión Federal de Telecomunicaciones.
<b>FG SSC</b>	<i>Focus Group on Smart Sustainable Cities</i> , Grupo de Enfoque sobre Ciudades Sostenibles Inteligentes.
<b>FSK</b>	<i>Frequency Shift Keying</i> , Modulación por Desplazamiento de Frecuencia.
<b>GFSK</b>	<i>Gaussian Frequency Shift Keying</i> , Modulación por Desplazamiento de Frecuencia Gaussiana.
<b>GMSK</b>	<i>Gaussian Minimum Shift Keying</i> , Modulación por Desplazamiento Mínimo Gaussiano.
<b>GPS</b>	<i>Global Positioning System</i> , Sistema de Posicionamiento Global.
<b>GSM</b>	<i>Global System for Mobile Communications</i> , Sistema Global de Comunicaciones Móviles.
<b>HTTP</b>	<i>HyperText Transfer Protocol</i> , Protocolo de Transferencia de Hipertexto.
<b>ICSP</b>	<i>In-Circuit Serial Programming</i> , Programación Serial en Circuito de Corriente alterna.
<b>ICT</b>	<i>Information and Communications Technology</i> , Tecnologías de la

	Información y las Comunicaciones.
<b>IEEE</b>	<i>Institute of Electrical and Electronics Engineers</i> , Instituto de Ingenieros Eléctricos y Electrónicos.
<b>IoT</b>	<i>Internet of Things</i> , Internet de las Cosas.
<b>ISO</b>	<i>International Organization for Standardization</i> , Organizaciones como la Organización Internacional de Estandarización.
<b>ITS</b>	<i>Intelligent Transportation System</i> , Sistemas de Transporte Inteligente.
<b>ITU</b>	<i>International Telecommunication Union</i> , Unión Internacional de Telecomunicaciones.
<b>I2I</b>	<i>Infrastructure to Infrastructure</i> , Infraestructura a Infraestructura
<b>I2V</b>	<i>Infrastructure to Vehicle</i> , Infraestructura a Vehículo.
<b>LCD</b>	<i>Liquid Crystal Display</i> , Pantallas de Cristal Líquido.
<b>LDPC</b>	<i>Low Density Parity Check-Code</i> , Codificación de Paridad de Baja Densidad.
<b>LoRa</b>	<i>Long Range</i> , Largo Alcance.
<b>LPWAN</b>	<i>Low Power Wide Area Network</i> , Redes de Área Amplia de Baja Potencia.
<b>MaB</b>	Modulo a Bordo.
<b>MG</b>	Modulo Gateway.
<b>MIMO</b>	<i>Multiple Input Multiple Output</i> , Múltiples Entradas y Múltiples Salidas.
<b>MP</b>	Módulo parada.
<b>MPT 1327</b>	<i>Ministry of Posts and Telegraph</i> , Ministerio de Correos y Telégrafos.
<b>MS</b>	Modulo Servidor.
<b>MS</b>	<i>Mobile Station</i> , Estaciones Móviles.
<b>OFDMA</b>	<i>Orthogonal Frequency-Division Multiple Access</i> , Acceso Múltiple por División de Frecuencias Ortogonales.
<b>PITS</b>	<i>Public Intelligent Transport System</i> , Sistemas de Transporte Público Inteligente.
<b>PoE</b>	<i>Power over Ethernet</i> , Alimentación a través de Ethernet.
<b>PMR</b>	<i>Private Mobile radio</i> , Radio Móvil Privado.
<b>PWM</b>	<i>Pulse Width Modulation</i> , Modulación por Ancho de Pulsos.
<b>QAM</b>	<i>Quadrature Amplitude Modulation</i> , Modulación de Amplitud en Cuadratura.
<b>SC-FDMA</b>	<i>Single Carrier Frequency Division Multiple Access</i> , Acceso Múltiple por División de Frecuencia de Portadora Única.

<b>SF</b>	<i>Spreading Factor</i> , Factor de ensanchamiento.
<b>SRWT</b>	<i>Short Range Wireless Technology</i> , Tecnologías Inalámbricas de Corto Alcance.
<b>TDMA</b>	<i>Time Division Multiple Access</i> , Acceso Múltiple por División de Tiempo.
<b>TETRA</b>	<i>Terrestrial Trunked Radio</i> , Radio Troncal Terrestre.
<b>UN</b>	<i>United Nations</i> , Naciones Unidas.
<b>UWB</b>	<i>Ultra Wide Band</i> , Banda Ultra Ancha.
<b>V2V</b>	<i>Vehicle to Vehicle</i> , Vehículo a Vehículo.
<b>WiSUN</b>	<i>Smart Utility Networks</i> , Redes de Utilidad Inteligentes.
<b>W-SIG</b>	<i>Weightless Special Interest Group</i> , Grupo de Interés Especial <i>Weightless</i> .

# 1. CAPÍTULO 1: INTRODUCCIÓN

Las ciudades antiguas se componían de estructuras físicas básicas, constituidas de techos y paredes, cuyo fin era brindar protección a sus habitantes. Con el paso del tiempo y de acuerdo a las necesidades que fueron apareciendo, se empezaron a incorporar a estas ciudades diferentes facilidades y sistemas, tales como: acueductos, sistemas de distribución de energía, sistemas de comunicaciones, infraestructura de transporte vial (calles y puentes), entre otros; todo ello, con el propósito de mejorar la calidad de vida de las personas [1].

El proceso de concebir, desarrollar e implementar dichos sistemas, llevó cientos de años. A pesar de todo lo avanzado en épocas anteriores, aún se siguen buscando diferentes alternativas para mejorar los procesos existentes y convertir las ciudades actuales, en ciudades sostenibles, eficientes e innovadoras [1].

La transformación de las ciudades de hoy, enfrenta el constante crecimiento territorial, ya que gran parte de la población rural se está desplazando hacia la zona urbana, en busca de mejores alternativas profesionales y de una mejor calidad de vida. Según la organización de las Naciones Unidas (UN, *United Nations*), el año 2008 marcó el periodo en que más del 50% de las personas del planeta (3.300 millones), vivían en zonas urbanas, cifra que se estima aumentará al 70% hacia el año 2050. Se identifica adicionalmente que alrededor del 80% del producto interno bruto mundial se genera en las ciudades, cifra que aumenta continuamente [2].

En este contexto, las ciudades actuales seguirán transformándose y organizándose en ecosistemas cada vez más complejos, afrontando el crecimiento de las necesidades de sus habitantes, generando a su vez, bienes y servicios que aumentan el impacto ambiental. Debido a esto, las ciudades se enfrentan a retos de tipo social, ambiental, organizativo, de gestión y económicos, que hacen que los gobiernos adopten las Tecnologías de la Información y las Comunicaciones TIC (ICT, *Information and Communications Technology*), como base para elaborar planes que permitan que las ciudades existentes y emergentes se transformen en Ciudades Inteligentes [3] [4].

## 1.1 CIUDADES INTELIGENTES

Las Ciudades Inteligentes hacen uso de las TIC en los diversos ámbitos urbanos para dar seguimiento a un desarrollo eficiente, y así, optimizar recursos, contribuir a la reducción del impacto ambiental y promover la actividad económica, con el fin de satisfacer las necesidades de sus habitantes.

En las Ciudades Inteligentes la inversión en capital humano, social y físico, genera un desarrollo económico sostenible, consecuencia de una adecuada y transparente gestión de los recursos económicos y naturales a través de un gobierno participativo [3].

Existen diversas definiciones de Ciudad Inteligente, y resulta difícil identificar una en común debido a las variables demográficas y geográficas que determinan el comportamiento de las mismas, sin embargo, la calidad de vida y el bienestar de las personas y del medio ambiente son el eje principal de este concepto. Para desarrollarse como Ciudad Inteligente es necesario, principalmente, realizar cambios en las diferentes infraestructuras que soportan el quehacer y desarrollo de sus habitantes: vías, edificios, servicios, redes, entre otros, y en la percepción que ellos tienen de la misma, logrando así, una combinación entre las personas y el entorno, con el fin de crear conciencia en los ciudadanos del buen uso de los nuevos servicios y recursos [3] [5].

Como se mencionó anteriormente, para lograr que una ciudad sea inteligente se requiere la integración de las TIC en los hábitos y costumbres de sus ciudadanos y en los espacios que usan de la misma, llevándolos a desarrollar y a hacer uso de sistemas sofisticados, conformados por redes, sistemas de información y plataformas de comunicación, en los que se procesa e integra un gran volumen de información en tiempo real, los cuales mejoran las operaciones y servicios en los entornos social, económico, ambiental y político que componen la ciudad [4].

Esto genera una participación más efectiva y activa de los ciudadanos, mejora la organización y gestión de la ciudad y el bienestar de quienes la conforman, reduce los costos operativos y el consumo de recursos, generando nuevas oportunidades de negocio, que hace más atractivo el vivir y hacer parte de una Ciudad inteligente [5].

Según Harrison, el término “Ciudad Inteligente” denota una ciudad “*instrumentada, interconectada e inteligente*” [6]. *Instrumentada* se refiere a la capacidad de capturar e integrar datos del mundo real, a través del uso de sensores, medidores, aparatos electrodomésticos, dispositivos personales, entre otros. *Interconectada*

significa la integración de los datos recolectados en una plataforma informática, que permite la comunicación y el intercambio de información entre los distintos servicios urbanos y los habitantes. *Inteligente* se refiere a la inclusión de servicios complejos de análisis, modelado, optimización y visualización de datos para tomar mejores decisiones operativas a nivel gubernamental y administrativo [6].

Las Ciudades Inteligentes se han convertido en tópico de investigación y análisis a nivel mundial, y se considera que son espacios urbanos con características innovadoras que agrupan las siguientes dimensiones: *Personas*, en términos de descubrir y satisfacer sus necesidades actuales y futuras; *Vida*, mejorar la calidad de vida y la coherencia social, así como la eficiencia energética, alimentaria e hídrica; *Medio ambiente*, protección, control de residuos y emisiones, y resistencia al cambio climático; *Gobernanza*, en términos de garantizar la disponibilidad de servicios fundamentales; *Economía*, crecimiento sostenible y competitividad de las ciudades; *Movilidad*, abordar los problemas de transporte, desplazamiento y gestión del tráfico [7].

#### 1.1.1. Desarrollo de una Ciudad Inteligente

Existen dos categorías para el desarrollo de Ciudades Inteligentes: Proyectos de ciudades nuevas y procesos de transformación de ciudades [2].

Proyectos de ciudades nuevas, como su nombre lo indica, son ciudades que se construyen desde cero, lo que permite que todos los subsistemas sean correctamente concebidos, planeados y ajustados a las necesidades de sus habitantes. Como ejemplos de ciudades nuevas están: *Tianjin* (China), *Songdo* (Corea) y *Paju Si* (Corea). *Tianjin* se enfoca en el concepto de ciudad verde<sup>1</sup>, centrándose en el medio ambiente y en el uso eficiente de recursos; *Songdo* se enfoca en ser una ciudad de negocios, la cual se concibe como un espacio en donde visionarios urbanísticos generen una alta demanda de oportunidades, ambientes diversos e innovación, propicia para los negocios; *Paju Si*, se enfoca hacia la felicidad de sus habitantes, teniendo en cuenta principalmente su participación, creando un modelo de ciudad pionera en educación y transporte [2].

Los procesos de transformación de ciudades se encargan de la optimización de los servicios e infraestructuras existentes, innovando en aspectos organizativos, funcionales, culturales y sociales. Un proceso de transformación puede llevar muchos años, y depende de la voluntad de sus gobernantes, de sus ciudadanos y de las inversiones a realizar. Ejemplos de ciudades en proceso de transformación son: Ciudad de Singapur (Singapur), Curitiba (Brasil) y Ciudad de Luxemburgo

---

<sup>1</sup> Las ciudades verdes fomentan la funcionalidad, la ecología y la belleza en sus infraestructuras para promover la calidad ambiental del entorno urbano.

(Luxemburgo); el éxito de ellas, se debe a que han logrado integrar exitosamente en sus políticas la actividad económica, la sostenibilidad ambiental y la calidad de vida de sus habitantes [2].

### 1.1.2. Estandarización para una Ciudad Inteligente

El desarrollo del concepto de Ciudades Inteligentes se respalda por estándares establecidos por los principales organismos de normalización a nivel global, los cuales son aceptados y permiten soluciones totalmente interoperables que pueden implementarse y replicarse a escala.

Organizaciones como la Organización Internacional de Estandarización (ISO, *International Organization for Standardization*), crea estándares a nivel mundial, que juegan un papel importante en el desarrollo de Ciudades Inteligentes, dentro de dichos estándares se encuentra específicamente el ISO 37122, que define indicadores de rendimiento para medir y considerar aspectos que aumentan el ritmo al que las ciudades mejoran sus resultados de sostenibilidad social, económica y ambiental, respondiendo a desafíos tales como el cambio climático, el rápido crecimiento de la población y la inestabilidad política y económica, mejorando fundamentalmente la forma en que las ciudades se involucran en la sociedad [8] [1].

El Instituto de Ingenieros Eléctricos y Electrónicos (IEEE, *Institute of Electrical and Electronics Engineers*), ha desarrollado estándares para Ciudades Inteligentes y sus diferentes componentes, entre ellos el Internet de las Cosas (IoT, *Internet of Things*) y los Sistemas de Transporte Inteligente (ITS, *Intelligent Transportation System*).

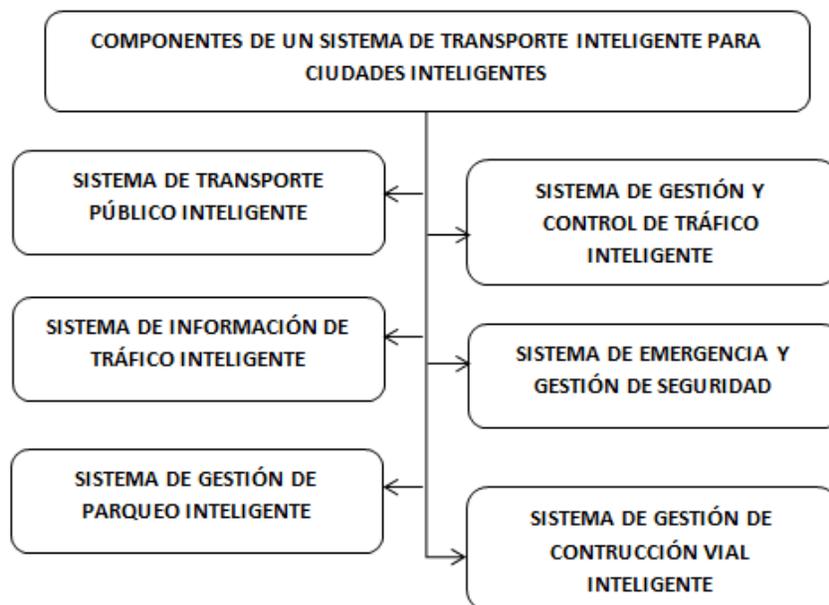
La Unión Internacional de Telecomunicaciones (ITU, *International Telecommunication Union*), establece un nuevo Grupo de Enfoque sobre Ciudades Sostenibles Inteligentes (FG SSC, *Focus Group on Smart Sustainable Cities*), encargado de evaluar los requisitos globales de estandarización de Ciudades Inteligentes. FG SSC actúa como una plataforma abierta para ciudades inteligentes, logrando identificar los estándares necesarios para apoyar la integración de los servicios de las TIC en cada ciudad [9].

## 1.2. SISTEMAS DE TRANSPORTE INTELIGENTE (ITS)

Los Sistemas de Transporte Inteligente se basan en las comunicaciones Vehículo a Vehículo (V2V, *Vehicle to Vehicle*), Infraestructura a Vehículo (I2V, *Infrastructure to Vehicle*) e Infraestructura a Infraestructura (I2I,

*Infrastructure to Infrastructure*<sup>2</sup>, para hacer el intercambio de información recolectada a través de sensores y equipos desplegados en vehículos e infraestructuras; este intercambio de información proporciona servicios que tienen como objetivo mejorar el sistema de transporte actual, haciéndolo más eficiente, sostenible, seguro y respetuoso con el medio ambiente [10].

Debido a la rápida urbanización, las ciudades presentan diferentes problemáticas con respecto a la movilidad, como consecuencia de la alta existencia de transporte privado, unido a las opciones de transporte público. Diferentes alternativas se han desarrollado y aplicado, con el fin de mitigar dichos problemas, a los cuales se suman la inseguridad vial, el alto consumo de combustible y el impacto ambiental. Algunas de estas medidas han aportado tanto factores positivos como negativos; tomando como ejemplo el “pico y placa” aplicado en muchas ciudades de Colombia, se reduce en cierta medida la congestión de tráfico y da paso a una movilidad más fluida, pero en su defecto, al restringir el uso de un vehículo uno o dos días a la semana, muchos ciudadanos optan por adquirir dos o más de estos para evadir dicha medida, lo que genera una situación perjudicial para la movilidad y el medio ambiente. El ejemplo anterior fomenta la importancia en que los gobiernos locales y nacionales implementen ITS basados en TIC, los cuales, pueden mejorar significativamente la eficiencia de los viajes, reducir los incidentes de tráfico, mejorar la seguridad, disminuir la congestión y reducir el impacto ambiental en las ciudades [1] [11]. La *Figura 1.1* presenta los diferentes componentes de un Sistema de Transporte Inteligente.



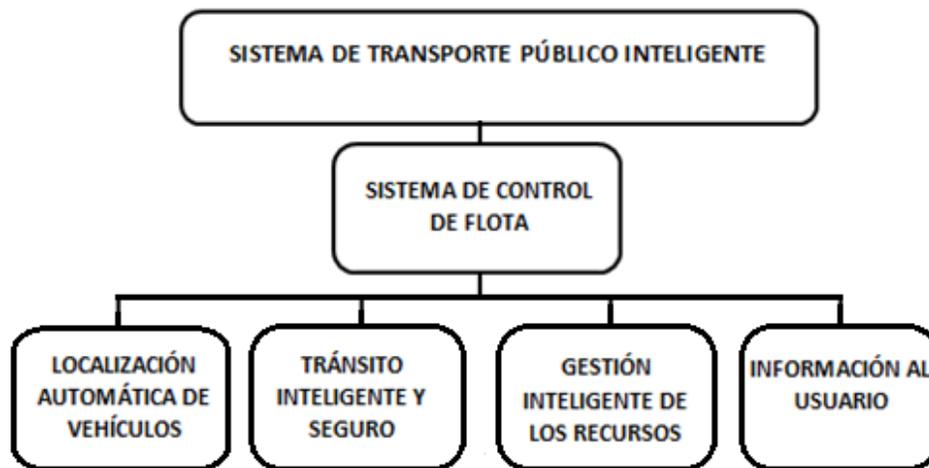
**Figura 1.1 Componentes de un Sistema de Transporte Inteligente (Por los autores).**

<sup>2</sup> Según la notación correspondiente a la ITU, la cual utiliza la letra V de vehículo e I de infraestructura para referirse a sus escenarios.

### 1.2.1. Sistema de Transporte Público Inteligente (PITS)

Orientado a la movilidad de una Ciudad Inteligente, se encuentra el sistema de transporte público, que se oferta como la solución más apropiada para reducir la congestión del tráfico en ciudades con alta densidad de población. Al generar conciencia del uso del transporte público y transportar un numeroso grupo de personas en un mismo bus, se evidenciará en las calles la reducción de vehículos circulando simultáneamente. Para que el transporte público sea el transporte de preferencia de los ciudadanos, se debe ofrecer un servicio cómodo, seguro, rápido y confiable. En busca de una solución que genere un servicio más eficiente, se implementa el concepto de transporte inteligente, al cual se le denomina Sistemas de Transporte Público Inteligente (PITS, *Public Intelligent Transport System*) [1] [12].

Dentro del PITS, existe el sistema de control de flota, el cual, optimiza e innova aplicaciones y servicios, que mejoran la calidad del servicio ofrecido. El control de flota abarca 4 componentes necesarios para el buen desarrollo del servicio lo cuales se observan en la *Figura 1.2*.



*Figura 1. 2 Componentes del PITS (Por los autores).*

Los componentes del PITS observados en la *Figura 1.2* se describen a continuación:

- La localización automática de vehículos, permite a los vehículos de servicio público reportar su ubicación, mediante dispositivos que posean el Sistema de Posición Global (GPS, *Global Positioning System*), haciendo posible que el ente encargado tenga visión en tiempo real del estado de todos los activos en los PITS.

- El tránsito inteligente y seguro, se refiere principalmente a la movilidad eficiente, lo que permite a los usuarios tomar decisiones respecto a las rutas que desean transitar, para lograr esto, los PITS se conforman de dispositivos y sensores, tales como: cámaras, sensores de movimiento y de velocidad, entre otros, que aportan información correspondiente del estado de funcionalidad del sistema tanto al ente encargado como a los usuarios, para una óptima gestión del tráfico.
- La gestión inteligente de los recursos corresponde a una administración adecuada en los PITS, comprometiéndose a un proceso constante de mejora, esto se logra mediante diferentes tipos de software que realizan el seguimiento a cada proceso que enfrentan, para que se efectúen de la mejor manera.
- La información al usuario permite darles a conocer datos relevantes de los PITS en tiempo real, mediante paneles de información, mensajes de texto, páginas Web o aplicaciones móviles y por comunicación con un *call center*, esta información es útil para ahorrar tiempo de viaje de los usuarios, entre otros servicios [13].

Los anteriores componentes ofrecen a los usuarios nuevos servicios en sus infraestructuras, mejores capacidades de movilidad, mayor seguridad, logrando que los PITS sea la opción más llamativa para los viajeros.

### 1.3. SISTEMA DE CONTROL DE FLOTA

Como se mencionó anteriormente el sistema de control de flota busca mejorar la calidad del servicio ofrecido en los PITS. Entre las aplicaciones desarrolladas dentro del sistema de control de flota, se destaca la información brindada al usuario, puesto que es de vital importancia que los pasajeros cuenten con información constante, precisa, y oportuna del servicio.

El sistema de control de flota considera 4 procesos que permiten la obtención de la información necesaria para ser expuesta al usuario. *La colección de datos*, mediante dispositivos ubicados sobre las vías, dentro de los buses o datos suministrados por los usuarios; *la transmisión*, basándose en una tecnología radio que comunique dichos dispositivos captadores de información al centro de control; *análisis de datos*, necesario para brindar información de calidad, y *despliegue de información*, mediante dispositivos finales.

El proceso de recoger y analizar datos en tiempo real, también permite hacer seguimiento de los vehículos para proveer diferentes soluciones que optimicen los

servicios brindados, tales como: el control de velocidades, cuantificación de kilómetros efectuados en horarios específicos, identificación de conductas e infracciones de tránsito y brindar soluciones como: la reducción de tiempos en los recorridos, optimización de los viajes, reducción del consumo de combustible, minimización de riesgos de accidentes, mantenimiento adecuado, localización del vehículo en cualquier momento, entre otras [14].

Para realizar el seguimiento de los vehículos utiliza tecnologías inalámbricas y móviles y sistemas integrados, que permiten la planificación de rutas, logrando el seguimiento y la localización de vehículos, gestionando rutas para el control de tráfico, adaptando la ruta a situaciones especiales, también controlando consumos y usos de la flota para ahorrar costos. Para ello es necesario dotar a los vehículos de una serie de equipos sensores y localizadores, con elementos de captación de información y un soporte de *software* para el monitoreo que puede adaptarse a los ya existentes en las ciudades, para controlar la base de datos con diversa información relativa a la situación de los destinos y servicios en la ruta, velocidad del móvil, identificación del conductor, entre otros datos útiles [15].

Aprovechando los sistemas de comunicación y de posicionamiento satelital, es posible mejorar la precisión en los horarios y frecuencias de las rutas para disminuir los tiempos de espera de los usuarios; esto se puede lograr con información confiable en tiempo real.

En la recopilación de información también es muy importante la contribuida por los usuarios; estos deben participar de forma activa para que sean involucrados en los procesos de planificación e información de un PITS, permitiendo la recopilación de datos a bordo; de este modo, se necesitan interfaces gráficas entre vehículos, infraestructura, conductores y pasajeros [16].

#### 1.3.1. Información al usuario

La información que se despliega en los dispositivos finales y es proporcionada al usuario, puede ser de interés sobre el viaje que el usuario desea realizar como: tiempos de espera, rutas, próxima parada, disponibilidad de asientos, entre otros. Opcionalmente se puede desplegar datos informativos del entorno como incidentes en la vía, desvíos, información sobre el conductor o la entidad que presta el servicio de transporte, estados de los vehículos o información sobre prevención, concientización y seguridad vial.

## **2. CAPÍTULO 2: ANÁLISIS DE TECNOLOGÍAS RADIO PARA LA TRANSMISIÓN DE INFORMACIÓN EN UN SISTEMA DE CONTROL DE FLOTA**

En este capítulo se realiza el análisis de algunas tecnologías radio, que ofrezcan facilidades para llevar a cabo la transmisión de información al usuario en un sistema de control de flota, bajo el marco de Ciudad Inteligente, determinando así, la más apropiada para su uso en el presente trabajo.

### **2.1. PARÁMETROS DE ANÁLISIS**

Para lograr la transmisión de datos en el escenario establecido y la realización de pruebas, se requiere de tecnologías radio que cuenten principalmente con los siguientes parámetros:

- ✓ Cobertura amplia: Se requiere de una tecnología cuya cobertura abarque áreas extensas (mayor a un kilómetro).
- ✓ Costo reducido: Es necesario un costo moderado en dispositivos que harán parte del piloto y que sean fácilmente financiables, además la tecnología debe operar en bandas no licenciadas con el fin de reducir costos de operación.
- ✓ Frecuencias bajas: Se requiere trabajar con una tecnología que opere en frecuencias bajas para que pueda garantizar un mayor alcance (se consideran las bandas Sub-GHz como frecuencias bajas).
- ✓ Escalabilidad: Es apropiado que la tecnología pueda adaptarse a cambios sin perder la calidad, o bien, manejar adecuadamente el crecimiento continuo de la red.
- ✓ Baja latencia: Es necesario que la tecnología maneje un retardo tolerable que no afecte el propósito del piloto.

### **2.2. ANÁLISIS DE TECNOLOGÍAS RADIO**

Existen diversas tecnologías que permiten la comunicación entre dispositivos; teniendo en cuenta los parámetros mencionados en el ítem 2.1, se hace un

análisis inicial, para elegir el grupo de tecnologías que cumpla con las características más convenientes para el piloto.

- Tecnología móvil celular

La tecnología móvil celular permite la comunicación inalámbrica entre dispositivos sin la necesidad de terminales fijos; la comunicación se establece mediante ondas de radio que efectúan cada una de sus operaciones, entre las cuales se encuentran: el envío de mensajes de texto, llamadas, conexiones a Internet, entre otras aplicaciones funcionales y de entretenimiento [17] [18].

En la tecnología móvil, las zonas de servicio se dividen en celdas, cada una de estas, funciona mediante una Estación Base (*BS, Base Station*) y Estaciones Móviles (*MS, Mobile Station*) que se mueven a través de celdas; cada MS es atendido por la BS que le corresponde, sin que se pierda la comunicación establecida, garantizando movilidad mediante procesos de *Handover* (transfiere el servicio de una BS a otra cuando la calidad del enlace es insuficiente en una de las estaciones, sin intervención del usuario) y *Roaming* (permite usar las redes de otros operadores fuera del país). Los sistemas actuales de telefonía móvil operan en bandas de frecuencias licenciadas (bandas de 800 MHz, 900 MHz, 1800 MHz, 1900 MHz, entre otras) definidas por las diferentes normas de transmisión de cada país. Los sistemas móviles celulares regularmente pueden brindar gran cobertura dependiendo del diseño que se tenga y de la densidad de usuarios; a medida que aumenta la cobertura el costo es mayor, es una tecnología escalable, de baja latencia y con soporte para un gran número de dispositivos [13] [17] [18].

- Radio Trunking

*Radio trunking* es un sistema de comunicación inalámbrico inmediato, es decir, permite conexión entre dos personas o un conjunto de personas al oprimir un botón [16]. En los sistemas convencionales de radio, las conversaciones son realizadas mediante una frecuencia dedicada; para *Radio trunking* las conversaciones entre dos personas o un grupo de personas se realizan a través de la asignación dinámica de frecuencias de las bandas de operación a medida que van siendo solicitadas por el usuario [19] [20].

En *Radio trunking* al ser un sistema troncalizado, el manejo de las vías de comunicación se hace por medio de un control central, asignando las frecuencias a cada canal; la asignación de frecuencias o canales se hace a través de diferentes protocolos de señalización analógicos y digitales, tales como: Radio Troncal Lógica (*LTR, Logic Trunked Radio*), Sistema de Comunicaciones de Acceso Digital Mejorado (*EDACS, Enhanced Digital Access Communications*

*System*), Radio Troncal Terrestre (TETRA, *Terrestrial Trunked Radio*), Ministerio de Correos y Telégrafos (MPT 1327, *Ministry of Posts and Telegraph*), Red Mejorada Digital Integrada (iDEN, *Integrated Digital Enhanced Network*), y *Motorola trunking* [17] [20].

Diferentes sistemas de transporte público en Colombia como el Metro de Medellín y el Sistema Estratégico de Transporte Público de Pasajeros para la ciudad de Pasto (AVANTE SETP), implementan como tecnología de comunicación *Radio trunking*, específicamente el estándar TETRA, el cual establece la comunicación entre vehículos y el centro de control, bajo un rango de cobertura bastante amplio [13] [18] [19] [21].

#### ➤ TETRA

TETRA es una tecnología radio definida a través de un estándar de modulación digital, creado por el Instituto Europeo de Estandarización de las Telecomunicaciones (*ETSI, European Telecommunications Standards Institute*), con el propósito de estandarizar los sistemas de comunicación de formato digital para sistemas de Radio Móvil Privado (*PMR, Private Mobile radio*) [22].

TETRA utiliza una técnica de Acceso Múltiple por División de Tiempo (TDMA, *Time Division Multiple Access*) y permite el establecimiento de distintas formas de comunicación seguras, ya sea individual o grupal soportando un buen número de usuarios con cobertura nacional, local o regional. En general, los dispositivos TETRA están disponibles en varias bandas de frecuencia, desde 299 MHz hasta 800 MHz; TETRA para Colombia opera en la banda de 415 – 430 MHz [23]. Las bandas mencionadas son licenciadas en Colombia lo cual aumenta su costo [24] [19] [22].

#### • LPWAN

Las Redes de Área Amplia de Baja Potencia (*LPWAN, Low Power Wide Area Network*), son un tipo de redes de comunicación inalámbrica diseñadas para permitir comunicaciones de largo alcance a una velocidad de transmisión de datos baja. Se desarrollan para suplir necesidades de IoT; donde se busca interconectar una gran cantidad de dispositivos para recopilar e intercambiar datos, permitiendo el monitoreo remoto de los dispositivos a través de la utilización de servicios en la nube [25].

La cobertura de las LPWAN alcanza distancias del rango de los kilómetros; se caracterizan por operar en las bandas Sub-GHz sin licencia, permitiendo una

comunicación robusta y confiable; maneja pequeñas cantidades de datos de transmisión y un bajo consumo energético de las baterías de los dispositivos, garantizando una larga duración, por ello, manejan bajo costo de dispositivos e infraestructura. Las LPWAN son escalables y prometen ser compatibles con servicios tolerantes al retardo [26] [27] [28].

- SRWT

Las Tecnologías Inalámbricas de Corto Alcance (*SRWT, Short Range Wireless Technology*) son comunicaciones que abarcan rangos de centímetros a cientos de metros, las tecnologías más conocidas bajo esta descripción son: *Bluetooth*, Banda Ultra Ancha (*UWB, Ultra Wide Band*), *ZigBee* y *WiFi* [29] [30].

El área de cobertura máxima de las SRWT se aproxima a los 100 metros; manejan un bajo consumo de energía en sus dispositivos garantizando larga vida para estos. La comunicación es simple, de baja latencia y de bajo costo, permitiendo una gran cantidad de nodos. Son tecnologías escalables, generalmente operan en altas bandas de frecuencia, entre 2.4GHz – 5GHz (a excepción de *Zigbee* que opera en 868 MHz y 915 MHz), razón por la cual la cobertura es limitada [29] [30].

- Otras tecnologías

Existen diversos dispositivos, operando en diferentes frecuencias, creados por empresas que a su vez trabajan con estándares propietarios, lo cual restringe la interoperabilidad con otros dispositivos como es el caso de *Ubiquiti Networks*. Se hace el análisis correspondiente a esta tecnología por la disponibilidad de equipos *Ubiquiti* en la Universidad del Cauca.

- *Ubiquiti Nano Station Loco* 900Mhz

*Ubiquiti Networks* es una empresa estadounidense encargada de fabricar dispositivos para comunicaciones inalámbricas. El dispositivo *Ubiquiti Nano Station Loco* 900Mhz es fabricado por *Ubiquiti Networks* para enlaces de comunicación inalámbrica en exteriores, incluye una antena de Múltiples Entradas y Múltiples Salidas (MIMO, *Multiple Input Multiple Output*).

El dispositivo *Ubiquiti Nano Station Loco* de 900Mhz, opera en dicha banda, alcanzando aproximadamente 5 Km de cobertura en entornos urbanos. Cuenta con una velocidad de transmisión de datos de 150Mbps y utiliza el protocolo TDMA AirMax (propietario de *Ubiquiti Networks*), permitiendo la conexión de más de 300 usuarios por equipo [31].

En la *Tabla 2.1* se evidencia el análisis correspondiente de los parámetros de interés de los grupos de tecnologías anteriormente mencionados.

PARAMETROS	TECNOLOGIAS				
	Móviles	Radio Trunking (TETRA)	LPWAN	SRWT	Otras tecnologías (Ubiquiti)
Alcance	Alto	Alto	Alto	Bajo	Alto
Costo	Alto	Alto	Bajo	Bajo	Bajo
Frecuencia	A/B	Bajo	Bajo	Alto	Bajo
Escalabilidad.	Alto	Alto	Alto	Alto	Bajo
Latencia	Bajo	Bajo	Medio	Bajo	Bajo

*Tabla 2.1 Análisis grupos de tecnologías radio*

El conjunto de tecnologías que cumple con las mejores características teniendo en cuenta los parámetros de interés analizados, es LPWAN. A continuación, se describen las principales tecnologías correspondientes a LPWAN.

#### 2.2.1. Tecnologías LPWAN

- RPMA

*Ingenu*, (anteriormente conocido como: *On-Ramp Wireless*), es la organización encargada del desarrollo de la tecnología de Acceso Múltiple por Fase Aleatoria (RPMA, *Random Phase Multiple Access*), la cual brinda una comunicación de bajo costo, robusta y bidireccional [26].

RPMA está desarrollado especialmente para comunicaciones de Máquina a Máquina (M2M, *Machine To Machine*) e IoT, estableciendo bajas exigencias de energía, que permiten a los dispositivos una vida útil de más de 10 años; haciéndola ideal para comunicaciones remotas y para dispositivos que no están conectados a una red eléctrica [32]. En contraste con otras tecnologías LPWAN, RPMA no opera en las bandas de los Sub-GHz, opera en la banda Industrial, Científica y Médica (ISM, *Industrial, Scientific and Medical*) a 2.4 GHz. El uso de esta banda está abierto en el mundo sin necesidad de licencia, por lo cual, es más sensible a pérdidas e interferencias. Ingenu propone la instalación de dos antenas

en cada uno de sus dispositivos, para regular la problemática de interferencia generada por el tamaño reducido de la longitud de onda de las antenas, debido a que operan en una frecuencia alta [33] [25] [28].

RPMA ofrece amplias zonas de cobertura, 15 Km aproximadamente en entornos urbanos, empleando altas potencias de transmisión u optimizando la sensibilidad del receptor y la ganancia de la antena [34].

- NB Fi

Fidelidad de Banda Estrecha (NB-Fi, *Narrowband Fidelity*), es una tecnología de comunicación inalámbrica caracterizada por brindar comunicaciones de baja potencia y áreas amplias; *Waviot* patentó esta tecnología especialmente para comunicaciones M2M [35].

NB-Fi opera en las bandas libres ISM Sub-GHz (compatible con las bandas 915 MHz, 868 MHz y 433 MHz y con otros espectros libres de licencia Sub-GHz); por ser un protocolo abierto, es de fácil implementación en todo el mundo y permite la interoperabilidad entre diferentes fabricantes [35]. NB-Fi tiene un esquema de modulación por Desplazamiento de Fase Diferencial Binaria (DBPSK, *Differential Binary Phase Shift Keying*) y maneja una velocidad de transmisión de datos entre 10bps y 100bps. La cobertura de esta tecnología permite alcanzar más de 16 Km en entornos urbanos y aproximadamente 50 km en entornos rurales, operando con una topología en estrella [36] [28].

- WEIGHTLESS

El Grupo de Interés Especial *Weightless* (W-SIG, *Weightless Special Interest Group*) es la organización encargada de administrar la tecnología *Weightless*, la cual es de estándar abierto y ofrece amplias áreas de cobertura con un reducido costo; esta tecnología define 3 estándares [33] [28]

*Weightless-N*: Ofrece comunicación unidireccional, operando en las bandas libres ISM Sub-GHz, bajo el esquema de modulación digital DBPSK. La cobertura de esta tecnología alcanza aproximadamente 3 km en áreas urbanas y cuenta con una velocidad de transmisión de datos de 100 bps [26] [33] [28] [27].

*Weightless-W*: A diferencia del estándar N, ofrece comunicación bidireccional y opera en el espectro libre en las bandas de televisión (400 – 800 MHz), bajo el esquema de Modulación de Amplitud en Cuadratura (16-QAM, *Quadrature Amplitude Modulation*) o el esquema de Modulación por Desplazamiento de Fase

Binario (BPSK, *Phase Shift Keying Binary*). La cobertura de esta tecnología alcanza aproximadamente 5 km en entornos urbanos y cuenta con una velocidad de transmisión de datos entre 1 kbps y 10 Mbps [28] [33].

*Weightless-P*: Brinda comunicación bidireccional y opera en las bandas libres ISM Sub-GHz, bajo el esquema de Modulación por Desplazamiento Mínimo Gaussiano (GMSK, *Gaussian Minimum Shift Keying*) [28]. La cobertura del estándar P se reduce en comparación a los otros estándares, ya que este, alcanza áreas aproximadas a 2 km en entornos urbanos y maneja una velocidad de transmisión de datos entre 200 bps y 100 kbps [28] [33] [26] [27].

- TELENSA UNB

Telensa es una organización encargada de desarrollar soluciones para Ciudades Inteligentes, dentro de sus soluciones se encuentra Telensa Banda Ultra Estrecha (UNB, *Ultra Narrow Band*), es una tecnología de comunicación inalámbrica LPWAN, la cual opera en la banda ISM sub-GHz sin licencia. Como principal desventaja con respecto a otras tecnologías LPWAN, no admite comunicaciones en interiores [26] [37].

Esta tecnología ofrece una comunicación totalmente bidireccional, alcanza aproximadamente 2 km de cobertura en entornos urbanos y 4 km en entornos rurales. Se estima que una estación base que implemente esta tecnología, pueda conectarse hasta con 5000 nodos [33]. Telensa se basa en una topología en estrella y maneja una velocidad de transmisión de datos de 62.5 bps en enlace ascendente y 500 bps en enlace descendente. Telensa actualmente se enfoca en soluciones para Ciudades Inteligentes, como aplicación principal, se implementa para iluminación inteligente [38] [33] [25] [26] [27].

- LoRa

Largo Alcance (LoRa, *Long Range*), es una tecnología regulada por *LoRa Alliance*, una organización sin fines de lucro para el desarrollo de aplicaciones IoT, conformadas por diversas empresas de este sector. Su arquitectura se basa en una comunicación por medio de pasarelas, las cuales son la puerta hacia la plataforma IoT. [39]

Es una tecnología de Capa Física (PHY, *Physical Layer*) que se basa en una técnica de Modulación de Espectro Ensanchado de Chirp (*CSS, Chirp Spread Spectrum*), lo que conlleva a una mejor resistencia contra interferencias. LoRa se caracteriza por operar en las bandas ISM Sub-GHz, con un enlace de largo

alcance de hasta 5 km en zonas urbanas, ofrece una red de baja potencia, bajo costo, baja latencia y soporte para un buen número de dispositivos [40] [41].

El protocolo de capa superior con mejor soporte para LoRa es LoRaWAN, que está abierto y gestionado por *LoRa Alliance*. LoRaWAN admite comunicaciones bidireccionales seguras y móviles, que funciona en un modelo de suscripción sin costo para el operador con una implementación sencilla y lista para usar [40] [41].

- DASH7

*DASH Alliance Protocol* desarrollado por *DASH Alliance*, es una tecnología de comunicación dentro del ámbito de las LPWAN. Esta tecnología es diseñada para proporcionar conectividad de enlace, abarcando áreas amplias de cobertura con dispositivos de baja potencia [42].

La arquitectura de las redes DASH7 se compone de dispositivos finales, subcontroladores y pasarelas. Las pasarelas permanecen activas de forma continua, recogiendo datos de los dispositivos finales y a su vez, retransmitiéndolos al servidor. Los subcontroladores tienen la funcionalidad de las pasarelas, pero a menor potencia y con ciclos de reposo, siendo su función principal la de transmitir datos desde los puntos finales a las pasarelas. De esta manera, las redes DASH7 tienen una topología en árbol o, sin el uso de subcontroladores, en estrella [43] [33] [25] [42] [44].

DASH7 proporciona tres velocidades de transmisión de datos definidas: 9.6 Kbps, 55.555 Kbps y 166.667 Kbps, bajo el esquema de Modulación por Desplazamiento de Frecuencia Gausiana (GFSK, *Gaussian Frequency Shift Keying*) y opera en las bandas ISM Sub-GHz sin licencia, con un tamaño máximo de los paquetes de 256 bytes. En comparación con el resto de las tecnologías de comunicación LPWAN, DASH7 obtiene una menor latencia debido a su ciclo de funcionamiento, el cual comprueba el canal periódicamente, pero presenta un aumento en el consumo de energía. DASH7 permite un alcance de hasta 5 Km en entornos urbanos [43].

- EC-GSM-IoT

Cobertura Ampliada GSM-IoT (EC-GSM-IoT, *Extended Coverage-GSM-IoT*), es una tecnología celular LPWAN que opera en las bandas Sub-GHz licenciadas, permanece en continuo desarrollo por el Proyecto de Asociación de Tercera Generación (3GPP, *3rd Generation Partnership Project*), como una mejora al Sistema Global de Comunicaciones Móviles (GSM, *Global System for Mobile Communications*), reutilizando su diseño siempre que le sea posible [33] [45].

EC-GSM-IoT sigue los principios de GSM, pero haciendo modificaciones en los canales lógicos junto con una técnica de Acceso Múltiple por División de Código (CDMA, *Code Division Multiple Access*) superpuesta, esto con el fin de limitar la cantidad de señal de control requerida en la PHY del enlace descendente y aumentar la capacidad en la PHY del enlace ascendente, permitiendo que varios dispositivos transmitan en el mismo canal físico simultáneamente [45] [33]

EC-GSM-IoT proporciona cobertura GSM normal y cobertura extendida, se pueden admitir 50000 dispositivos por celda con una velocidad de transmisión de datos de 350 bps a 70 Kbps, dependiendo de la clase de cobertura [33].

- WiFi Hallow

El estándar IEEE 802.11ah, comercializado como *WiFi Hallow*, es un protocolo de comunicación inalámbrica de la PHY y la capa de Control de Acceso al Medio (MAC, *Media Access Control*) que adapta el IEEE 802.11ac para operar en las bandas de frecuencia ISM Sub GHz sin licencia. El objetivo de *WiFi Hallow* es proporcionar un mejor soporte para los sensores inalámbricos y aplicaciones IoT [43] [46].

Soporta una amplia gama de anchos de banda, esquemas de modulación y codificación tales como: Códigos Convolucionales Binarios (BCC, *Binary Convolutional Codes*) y Codificación de Paridad de Baja Densidad (LDPC, *Low Density Parity Check-Code*), lo cual, la convierte en una tecnología altamente flexible; permitiendo así, una velocidad de transmisión de datos de hasta 347 Mbps a corta distancia, y 150 Kbps a distancias máximas de 1 Km de cobertura [28] [27] [46].

- NB-IoT

Banda estrecha-IoT (NB-IoT, *Narrow Band IoT*) es una tecnología LPWAN celular desarrollada por el 3GPP definida en la versión 13, opera en una banda licenciada a diferencia de la mayoría de las redes LPWAN, que operan en las bandas sin licencia Sub-GHz [45].

Esencialmente, NB-IoT se construye a partir de LTE, manteniéndose lo más simple posible, en comparación a esta, cuenta con una velocidad de transmisión de datos más baja y una menor eficiencia espectral, con el fin de lograr mejor cobertura, reducir los costos del dispositivo y minimizar el consumo de la batería, por lo que simplifica muchas características de LTE [33] [47].

NB-IoT es bidireccional, el enlace descendente se basa en el Acceso Múltiple por División de Frecuencias Ortogonales (OFDMA, *Orthogonal Frequency-Division Multiple Access*) y el enlace ascendente se basa en el Acceso Múltiple por División de Frecuencia de Portadora Única (SC-FDMA, *Single Carrier Frequency Division Multiple Access*) [33] [25] [44].

NB-IoT puede desplegarse reutilizando y actualizando la red celular existente, sus despliegues no son adecuados para regiones rurales o suburbanas que no tienen cobertura 4G. El rango de cobertura que proporciona esta tecnología es hasta 35 Km en condiciones óptimas [25] [33].

- SigFox

SigFox es una tecnología propietaria de la empresa que lleva su propio nombre, desarrollada para soluciones de IoT. SigFox maneja diferentes esquemas de modulación en enlaces ascendentes y descendentes. Para el enlace descendente, utiliza un esquema de modulación GFSK con una velocidad de transmisión de datos de 500 bps en un canal de 600 Hz; para el enlace ascendente utiliza el esquema de modulación DBPSK con una transmisión de datos 100 bps en un canal de 100 Hz. Cada mensaje de enlace ascendente se transmite tres veces en diferentes frecuencias para asegurar que sea entregado [48].

La red SigFox opera sobre la tecnología de comunicación de Banda Ultra Estrecha (UNB, *Ultra Narrow Band*) tanto en su enlace ascendente como descendente, transmite sobre las bandas de frecuencias de Sub-GHz libres; 868 MHz en Europa y 915 en Estados Unidos, es bidireccional, con una cantidad limitada de mensajes de 140 mensajes en enlaces ascendentes y 4 mensajes en enlaces descendentes por día. El alcance de la tecnología varía según el entorno de aplicación, para zonas rurales tiene un rango de 30 Km a 50 Km y en zonas urbanas de 3Km a 10 Km [48] [49].

- WiSUN

*WiSUN Alliance* es una alianza industrial que se encarga de promover soluciones basadas en estándares inalámbricos interoperables para el IoT, entre su amplia gama de aplicaciones se encuentran, sistemas de estacionamiento y gestión de tráfico, por nombrar algunos ejemplos [50] [27].

El estándar IEEE 802.15.4g o también llamado Redes de Utilidad Inteligentes (WiSUN, *Smart Utility Networks*) se diseñó por *WiSUN Alliance* para lograr comunicaciones inalámbricas de largo alcance y baja potencia; opera

generalmente en las bandas de los Sub GHz y también en la banda libre de 2.4Ghz, permitiendo un área de cobertura aproximada a 1 Km. IEEE 802.15.4g se enfoca principalmente en el bajo consumo de energía, proporcionándole a los dispositivos una vida útil de aproximadamente 10 años; las velocidades de transmisión de datos oscilan entre 50 y 1000 kbps [27] [51].

### 2.2.1.1. Características técnicas de las tecnologías LPWAN

Se realiza un comparativo mediante las *Tablas 2.2 y 2.3* en las que especifican las principales características técnicas de las tecnologías LWPAN.

	RPMA	NB Fi	Weighthless			Telensa	LoRa
			N	W	P		
<b>Frecuencia</b>	2.4Ghz (ISM)	Sub Ghz ISM	Sub Ghz ISM	400-800 Mhz	Sub Ghz ISM	Sub Ghz ISM	Sub Ghz ISM
<b>Cobertura (Urbano)</b>	10-15Km	15Km	3Km	5Km	2Km	1Km	5Km
<b>Modulación</b>	DSSS-CDMA	DBPSK	DBPSK	16QAM, BPSK, QPSK, DBPSK	GMSK	UNB 2FSK	CSS
<b>MAC</b>	CDMA	-	ALOHA	TDMA, FDMA	TDMA, FDMA	-	ALOHA
<b>Tasa de datos UL</b>	624Kbps	10-100bps	100bps	1Kbps-10Mbps	200bps-100Kbps	62.5bps	300bps-50Kbps
<b>Tasa de datos DL</b>	156Kbps	10-100bps	No DL	1Kbps-10Mbps	200bps-100Kbps	500bps	300bps-50Kbps
<b>Topología</b>	Estrella, Árbol	Estrella	Estrella	Estrella	Estrella	Estrella	Estrella
<b>Nodos por Gateway</b>	500000	1350000	>1000000	>1000000	>1000000	5000	>1000000

**Tabla 2.2 Características técnicas de tecnologías LPWAN (1) [28] [30].**

	DASH7	EC-GSM-IoT	WiFi Hallow	NB IoT	SigFox	WiSUN
<b>Frecuencia</b>	Sub Ghz ISM	Sub Ghz Licenciada	Sub Ghz ISM	Sub Ghz Licenciada	Sub Ghz ISM	Sub Ghz y 2.4Ghz ISM
<b>Cobertura (Urbano)</b>	5Km	15Km	1Km	>15Km	10Km	>1Km
<b>Modulación</b>	GFSK	GMSK, 8PSK	BCC, LDPC	QPSK, OFDMA, SC-FDMA	UNB DBPSK, GFSK	FSK, OFDMA, OQPSK
<b>MAC</b>	CSMA/CA	GSM	-	LTE	ALOHA	-
<b>Tasa de datos UL</b>	9.6Kbps, 55.55Kbps o 166.667Mbps	350bps-70Kbps	150Kbps-346.66Mbps	50kbps	100 bps, 140 msj/día	50-1000Kbps
<b>Tasa de datos DL</b>	9.6Kbps, 55.55Kbps o 166.667Mbps	350bps-70Kbps	150Kbps-346.66Mbps	60kbps	Max 4 msj de 8 bytes/día	50-1000Kbps
<b>Topología</b>	Estrella, Árbol	Estrella	Estrella, Árbol	Estrella	Estrella	Estrella, Malla
<b>Nodos por Gateway</b>	-	50000	-	52000	>1000000	-

**Tabla 2.3 Características técnicas de tecnologías LPWAN (2) [28][30]**

Las características técnicas sirven para tener una idea más detallada del modo de operación de cada tecnología, sin embargo, la elección de la tecnología LPWAN, requerida para la transmisión de información al usuario en el presente trabajo de grado, se rige por los parámetros descritos en el ítem 2.1, a los cuales se adicionan los siguientes parámetros importantes a tener en cuenta:

- ✓ Movilidad: Es necesario que la tecnología cuente con características de movilidad que puedan dar soporte al sistema de control de flota.
- ✓ Transmisión continua de datos: Se requiere que la tecnología envíe constantemente información de ubicación.

En las *Tablas 2.4 y 2.5* se observa un análisis comparativo entre las tecnologías LPWAN.

	RPMA	NB Fi	Weigthless			Telensa	LoRa
			N	W	P		
<b>Cobertura</b>	Alto	Alto	Medio	Medio	Bajo	Bajo	Alto
<b>Costo</b>	Bajo	Bajo	Bajo	Bajo	Bajo	Bajo	Bajo
<b>Frecuencias</b>	Alto	Bajo	Bajo	Bajo	Bajo	Bajo	Bajo
<b>Escalabilidad</b>	Alto	Alto	Alto	Alto	Alto	Alto	Alto
<b>Latencia</b>	Medio	Medio	Medio	Medio	Medio	Medio	Medio
<b>Movilidad</b>	No	-	No	-	No	No	Si
<b>Transmisión continua de datos</b>	Alto	Alto	Alto	Alto	Alto	Alto	Alto

**Tabla 2.4 Comparativo entre tecnologías LPWAN (1)**

	DASH7	EC-GSM-IoT	WiFi Hallow	NB IoT	SigFox	WISUN
<b>Cobertura</b>	Alto	Alto	Bajo	Alto	Alto	Bajo
<b>Costo</b>	Bajo	Alto	Bajo	Alto	Bajo	Bajo
<b>Frecuencias</b>	Bajo	A/B	Bajo	A/B	Bajo	Bajo
<b>Escalabilidad</b>	Alto	Alto	Alto	Alto	Alto	Alto
<b>Latencia</b>	Medio	Medio	Medio	Medio	Medio	Medio
<b>Movilidad</b>	-	Si	-	No	No	-
<b>Transmisión continua de datos</b>	Alto	Alto	Alto	Alto	Bajo	Alto

**Tabla 2.5 Comparativo entre tecnologías LPWAN (2)**

Una vez analizadas las tecnologías, se observa que LoRa cuenta con las características más apropiadas para el desarrollo del presente trabajo de grado:

*Cobertura alta*, permitiendo que el área de cobertura sobrepase el escenario establecido; *Costo bajo*, logrando una posible adquisición de equipos que brinden soluciones de IoT; *Bandas de frecuencias bajas*, garantizando amplias áreas de cobertura; *Escalabilidad alta*, concibiendo el crecimiento de la red; *Latencia media* (característica de las LPWAN), admitiendo un retardo tolerable para el piloto; *Movilidad*, dando soporte de conexión a los dispositivos móviles y *Transmisión continua de datos*, garantizando el envío de la información constante al usuario.

## 3. TECNOLOGÍA LoRa

En este capítulo se presentan las principales características de la tecnología LoRa, dando a conocer su origen, modulación, parámetros, arquitectura, regulación e información conveniente para el desarrollo del presente trabajo de grado.

### 3.1. LoRa ALLIANCE

LoRa Alliance es una asociación abierta, conformada por fabricantes, desarrolladores y diseñadores que está en continuo funcionamiento y ha crecido a más de 500 miembros desde su creación en marzo de 2015, convirtiéndose en la alianza más grande y de mayor crecimiento en el sector de la tecnología. Tiene por objetivo estandarizar las redes de área extendida de baja potencia que se utilizan en todo el mundo para IoT, por lo cual, desarrolla, implementa y utiliza la tecnología en diferentes países, promoviendo la adopción global del protocolo LoRaWAN como el principal estándar abierto mundial para la conectividad segura de IoT, lo que garantiza la interoperabilidad de todos los productos y tecnologías LoRa [39].

### 3.2. LoRa Y LoRaWAN

#### 3.2.1. LoRa

LoRa es un esquema de modulación de espectro ensanchado derivado de la Modulación CSS, patentado por Semtech<sup>3</sup>, el cual usa pulsos de Chirp de frecuencia lineal de banda ancha, en donde la frecuencia aumenta o disminuye durante un tiempo para codificar información. LoRa Implementa un esquema de Velocidad de Datos Variable (*ADR, Adaptive Data Rate*), que utiliza Factores de ensanchamiento (*SF, Spreading Factor*) ortogonales, lo que permite intercambiar la velocidad de transmisión de datos por rango o potencia, a fin de optimizar el rendimiento de la red en un Ancho de Banda (*BW, Bandwidth*) constante [41] [40] [52].

---

<sup>3</sup> Semtech es un proveedor líder de semiconductores analógicos y de señal mixta de alto rendimiento y algoritmos avanzados para consumidores finales, informática empresarial, comunicaciones y mercados finales industriales.

La modulación empleada por Lora se caracteriza por tener propiedades como: BW escalable, uso de bajas potencias, alta robustez, largo alcance, resistencia al multitrayecto y al efecto Doppler. Al ser una implementación de PHY independiente de las implementaciones de capa superior, LoRa puede coexistir e interoperar con arquitecturas de red existentes, lo que le permite tener una gran escalabilidad [53] [52].

LoRa emplea una variación única de la modulación CSS, que incorpora cinco parámetros clave: Frecuencia, BW, SF, potencia de transmisión y Tasa de Codificación (CR, *Code Rate*); los cuales tienen influencia directa sobre el paquete, ya que su variación impacta directamente sobre la sensibilidad del receptor y la velocidad de transmisión de datos, lo que implica la creación de redes LoRa con diferentes coberturas a diferentes velocidades de transmisión [53] [54].

Actualmente LoRa admite siete factores de ensanchamiento diferentes que van desde SF6 hasta SF12, teniendo en cuenta que un mayor SF implica tener una red con velocidad de transmisión de datos más baja, pero con una mejor sensibilidad en el receptor obteniendo una mejor cobertura. El SF además de ser uno de los responsables de la cobertura y velocidad de transmisión, también proporciona la coexistencia de dispositivos LoRa, es decir, permite transmisiones simultáneas con diferentes SF en un mismo canal [55] [53] [54].

La *Tabla 3.1* muestra la influencia del ancho de banda en la sensibilidad del receptor y tiempo en aire<sup>4</sup> del paquete con un SF fijo de 10.

<b>BW (KHz)</b>	<b>Tiempo en aire (ms)</b>	<b>Sensibilidad del receptor (dBm)</b>
<b>125</b>	<b>264.2</b>	<b>-132</b>
<b>250</b>	<b>132.1</b>	<b>-129</b>
<b>500</b>	<b>66</b>	<b>-126</b>

**Tabla 3.1 Influencia del BW en el tiempo en aire y sensibilidad [54].**

La *Tabla 3.2* muestra la influencia del SF en la sensibilidad del receptor y tiempo en aire del paquete para un ancho de banda fijo de 250 KHz.

---

<sup>4</sup> Es el tiempo que se demora el paquete en llegar desde el lugar transmitido hasta su destino.

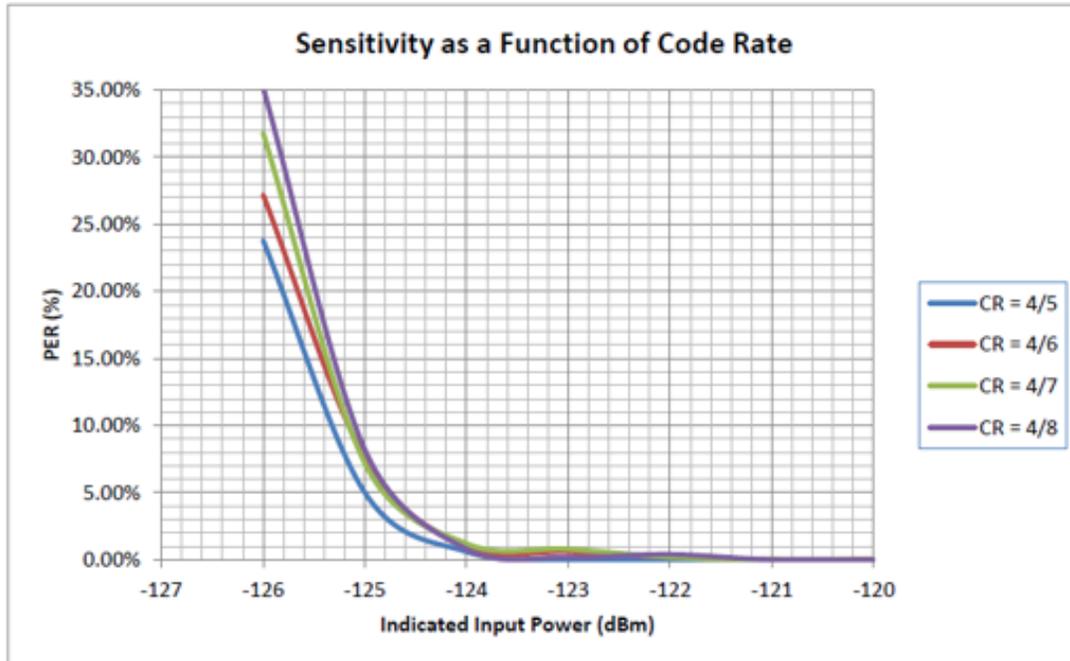
SF	Tiempo en aire (ms)	Sensibilidad del receptor (dBm)
12	528.4	-134
10	132.1	-129
8	39.2	-124

**Tabla 3.2 Influencia del SF en el tiempo en aire y sensibilidad [54].**

La CR en LoRa es el código de corrección de errores agregado a un paquete antes de la transmisión de datos, usa el Código *Hamming* como el código de corrección de errores hacia adelante, que permite la recuperación de bits de información debido a interferencias, el cual agrega una pequeña sobrecarga de codificación adicional de los datos en el paquete transmitido. En la *Tabla 3.3* se observa el impacto que tiene la CR sobre el tiempo en aire de los datos, para un BW fijo de 250 KHz y un SF de 10. Dependiendo de la CR seleccionada, la robustez adicional obtenida en presencia del ruido térmico se muestra en la *Figura 3.1*. [54] [53] [55].

CR	Tiempo en aire (ms)
1	123.9
2	132.1
4	148.5

**Tabla 3.3 Influencia de la Tasa de Codificación en el tiempo en aire [54].**

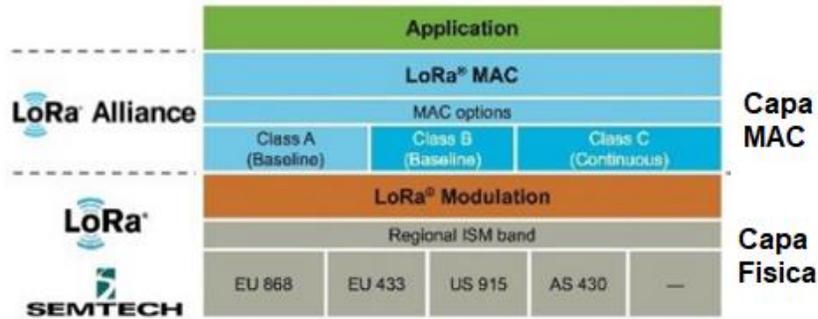


**Figura 3.1** Influencia de la Tasa de Codificación en la sensibilidad del receptor [54].

### 3.2.2. LoRaWAN

LoRaWAN es el protocolo de comunicación de control de acceso al medio y la arquitectura del sistema para la red, que permite la comunicación de muchos dispositivos mediante la modulación empleada por LoRa, controla el encendido y apagado de dispositivos finales proporcionando un buen manejo en la duración de la batería de los dispositivos, de igual forma LoRaWAN está enfocado a requisitos clave de IoT como la bidireccionalidad, servicios de comunicación, seguridad extremo a extremo, movilidad y localización [56]. Si bien la modulación de LoRa es propietaria, LoRaWAN es un estándar abierto desarrollado por LoRa Alliance [41] [39] [40].

En la *Figura 3.2* se evidencia la pila de protocolos para la tecnología LoRa, la cual solo especifica dos niveles del modelo de Interconexión de Sistemas Abiertos (OSI, *Open System Interconnection*), PHY y Capa MAC; establece a la modulación LoRa en la PHY y al protocolo de comunicación LoRaWAN en la Capa MAC junto a los dispositivos finales LoRaWAN.



**Figura 3.2 Pila de protocolos para la tecnología LoRa [41].**

### 3.3. CLASES DE DISPOSITIVOS LoRaWAN

LoRaWAN utiliza tres clases de dispositivos finales, en donde cada clase abarca diferentes requisitos.

- Dispositivos finales bidireccionales de menor potencia (Clase A):

Permiten comunicaciones bidireccionales. La transmisión de enlace ascendente de cada dispositivo final es seguida por dos ventanas de recepción<sup>5</sup> de enlace descendente cortas; con este patrón de comunicación este dispositivo tiene el menor consumo de energía de todos los dispositivos finales LoRa. Este dispositivo debe ser soportado por todas las clases LoRa [57] [40].

- Dispositivos finales bidireccionales con Ranuras de recepción programadas (Clase B):

Además de ser dispositivos bidireccionales con ventanas de recepción como el Clase A, los dispositivos Clase B abren sus ventanas de recepción adicionales en horarios programados. Para que el dispositivo final abra su ventana de recepción a la hora programada, recibe una señal de señalización temporizada desde la pasarela; esto permite al servidor saber cuándo está escuchando el dispositivo final. [57] [40].

- Dispositivos finales bidireccionales con ranuras de recepción máximas (Clase C):

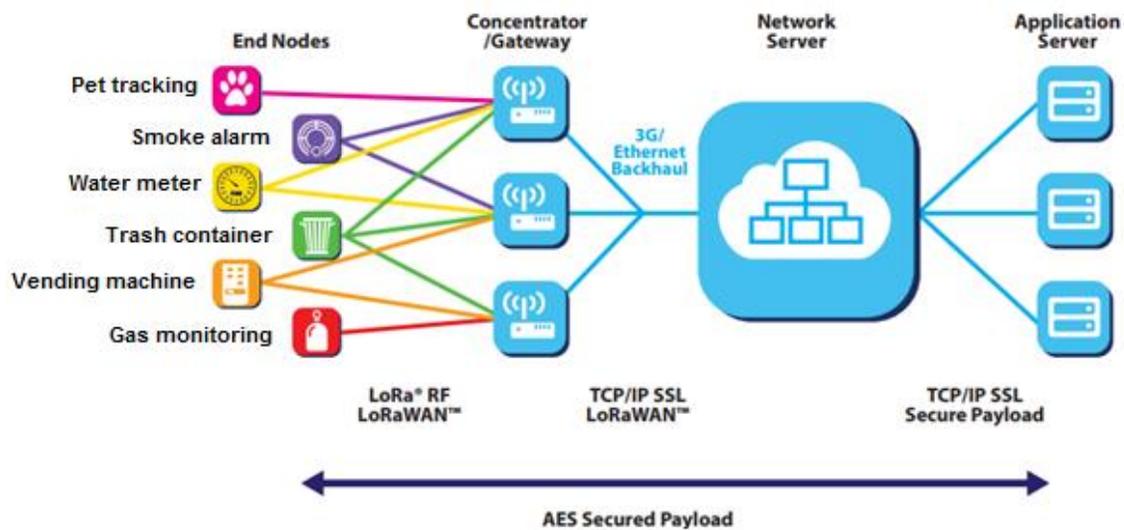
Son dispositivos que tienen ventanas de recepción continuamente abiertas, cerradas únicamente cuando el dispositivo está transmitiendo. Como el dispositivo

<sup>5</sup>Es la lista que tiene el receptor con los números de la secuencia consecutivos de las tramas que puede aceptar. Almacena las tramas temporalmente en un buffer hasta el momento de completarlas y mediante la lista, ordenarlas.

está continuamente escuchando requiere más consumo de energía que los de Clase A y Clase B, pero con menores tiempos de latencia [57] [40].

### 3.4. ARQUITECTURA DE LA RED LoRaWAN

LoRa normalmente se utiliza como una topología de estrella con el uso de pasarelas. La topología en estrella es simple, segura y optimiza el consumo de energía de los dispositivos finales, ya que puede comunicarse con la mejor pasarela, reduciendo así los tiempos de encendido de la unidad. La arquitectura de la red se observa en la *Figura 3.3*. [40] [57] [58].



**Figura 3.3** Arquitectura de la Red LoRaWAN [41].

La arquitectura de la red LoRaWAN consta de una o varias Gateway dependiendo de la cobertura y el tráfico de la red, dispositivos finales, un servidor de red y un servidor de aplicaciones. En una red LoRaWAN, los datos transmitidos por un dispositivo final (Clase A, Clase B y Clase C) son recibidos por la Gateway y de esta se reenvían a un servidor de red a través de la red celular, Ethernet o WiFi [53] [58].

### 3.5. SEGURIDAD LoRaWAN

LoRaWAN diseña específicamente la seguridad como un aspecto esencial, lo que conlleva a redes seguras preparadas para el futuro con mecanismos de seguridad como autenticación mutua, protección de integridad y confidencialidad; diseña su seguridad haciendo uso de algoritmos estándar y seguridad extremo a extremo,

ajustándose a los criterios de diseño de las redes LPWAN: bajo consumo de potencia, baja complejidad en la implementación, bajo costo y alta capacidad [59].

La seguridad en LoRaWAN usa el algoritmo criptográfico Estándar de Cifrado Avanzado<sup>6</sup> (AES, *Advanced Encryption Algorithm*) en conjunto de modos de operación como lo son: AES-Código de Autenticación de Mensaje Basado en Cifrado (AES-CMAC, *AES-Cipher-Based Message Authentication Code*)<sup>7</sup> para protección de integridad y Cifrado de Modo Contador<sup>8</sup> (CTR, *Counter Mode Encryption*) para cifrado. Cada dispositivo LoRaWAN tiene determinado una clave única AES de 128 bits llamada *AppKey* y un identificador único global *DevEUI* basado en EUI-64<sup>9</sup>, los cuales se utilizan durante los mecanismos de seguridad [59].

### 3.6. TRAMA DE COMUNICACIONES

La estructura de paquetes LoRa se divide en paquetes de enlace ascendente y descendente, lo que se indica por la presencia o ausencia del campo de Verificación de Redundancia Cíclica (CRC, *Cyclic Redundancy Check*) dispuesto al final de la trama LoRa, tiene un tamaño máximo de carga útil de 255 bytes y el número de símbolos a transmitir depende principalmente del SF [53] [55].

Una trama LoRa consta inicialmente de un preámbulo, en donde el último byte es una palabra de sincronización que se utiliza para diferenciar las redes que usan las mismas bandas de frecuencias; después del preámbulo hay un encabezado opcional que se desactivará en situaciones donde no es necesario, es decir, cuando se conoce la longitud de la carga útil, la CR y la presencia de CRC; el encabezado incluye un CRC para permitir que el receptor descarte paquetes con encabezados no válidos y finalmente le sigue la carga útil seguido de un CRC, el cual puede estar activado o desactivado dependiendo del tipo de paquete. En la *Figura 3.4* se observa la trama de comunicación de LoRa mencionada anteriormente [53] [55].

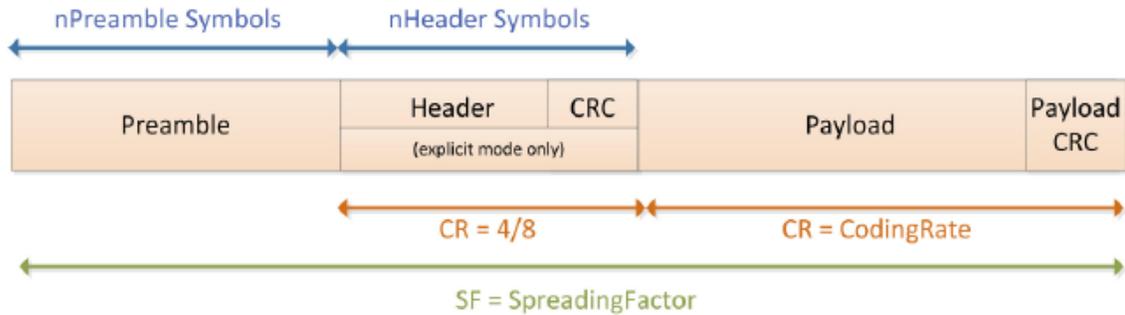
---

<sup>6</sup> AES es un algoritmo de cifrado público basado en claves secretas simétricas, que permite el cifrado y la autenticación de mensajes.

<sup>7</sup> AES-CMAC, es un código de autenticación de mensaje basado en cifrado que utiliza el algoritmo de cifrado AES para proporcionar integridad y autenticidad del mensaje.

<sup>8</sup> CTR, es un modo de operación del algoritmo AES que se basa en un contador para cifrar flujos de datos.

<sup>9</sup> Identificador Único Extendido-64 (EUI-64, *Extended Unique Identifier*), es un proceso que permite que un host le asigne un identificador único de interfaz IP de versión 6 de 64 bits.



**Figura 3.4 Trama de comunicación LoRa [54].**

### 3.7. REGULACIÓN LoRa

Al igual que todos los protocolos en las bandas ISM, LoRaWAN debe seguir las regulaciones definidas por la región en la que se implementa la red. A continuación, se muestra la regulación para Europa y para América del Norte, teniendo en cuenta que la regulación adoptada para Colombia es la de América del Norte [40] [60].

#### 3.7.1. LoRa para Europa

Europa define la banda ISM a 863-870 MHz, LoRaWAN cuenta con diez canales, ocho de los cuales son de múltiples velocidades de transmisión de datos de 250bps a 5.5 kbps, un solo canal LoRa de alta velocidad a 11kbps y un solo canal de Modulación por Desplazamiento de Frecuencia (*FSK, Frequency Shift Keying*) a 50kbps. La potencia de salida máxima permitida por ETSI en Europa es de +14dBm, con la excepción de la banda G3 (869.4 MHz – 869.64 MHz) que permite +27dBm. Hay restricciones de ciclo de trabajo bajo ETSI, pero no hay limitaciones máximas de transmisión ni de tiempo de permanencia del canal [57] [61] [40].

#### 3.7.2. LoRa para América Del Norte

La banda ISM para América del Norte es de 902-928MHz, en este estándar, LoRaWAN define 64 canales de enlace ascendente de 125 kHz de 902.3 a 914.9MHz en incrementos de 200 kHz. Hay ocho canales adicionales de 500KHz de enlace ascendente en incrementos de 1.6MHz de 903MHz a 914.9MHz. Los ocho canales de enlace descendente tienen un ancho de 500 kHz desde 923.3MHz hasta 927.5MHz. La potencia de salida máxima en la banda de 902-928MHz de América del Norte es de +30dBm, pero para la mayoría de los dispositivos es suficiente +20dBm. Bajo la Comisión Federal de Telecomunicaciones (*FCC, Federal Communications Commission*) no hay

limitaciones de ciclo de trabajo, pero hay un tiempo de permanencia máximo de 400 ms por canal [57] [61] [40].

## 4. CAPITULO 4: METODOLOGÍA DE DESARROLLO

La metodología del Proceso Unificado Racional (RUP, Rational Unified Process), es un proceso propietario de la ingeniería de software, creado por *Rational Software*, compañía de IBM. Es un proceso de desarrollo iterativo e incremental, en donde, el trabajo se divide en partes más pequeñas o mini proyectos; RUP consta de 4 fases de desarrollo: Requerimientos, Análisis y Diseño, Implementación y Pruebas, además cuenta con un proceso de retroalimentación, lo que permite que, una vez terminado el proyecto, regrese a la etapa inicial y verifique si requiere de alguna modificación [43].

El presente trabajo de grado implementa la metodología del Proceso Unificado Racional, para su adecuado desarrollo. En la *Figura 4.1*, se muestra el diagrama en bloques de la metodología RUP.



**Figura 4.1** Diagrama en bloques de la metodología RUP. Por los autores.

## 4.1. REQUERIMIENTOS

El piloto a implementar debe cumplir con los siguientes requisitos funcionales, los cuales están acorde a los objetivos de este trabajo de grado.

### 4.1.1. Requisitos funcionales

- Establecer la comunicación entre bus, Gateway y parada.
- Conocer el tiempo de llegada del bus a la parada.
- Desplegar la información en pantallas en cada parada, acerca del tiempo de llegada del bus y número de ruta.
- Mostrar ubicación del bus en tiempo real a través de una página web.
- Brindar información al usuario de fácil entendimiento.

### 4.1.2. Requisitos no funcionales

- El tiempo de espera visualizado por el usuario debe ser muy próximo al tiempo real.
- La información debe ser fiable.
- La tecnología elegida debe abarcar el área de cobertura correspondiente al escenario designado para el piloto.

### 4.1.3. Restricciones del sistema

- Cobertura brindada por la Gateway.
- Rango de error del GPS.
- Retardo de equipos.
- Línea de vista.
- Velocidad variable del bus.

## 4.2. ANÁLISIS Y DISEÑO

En esta sección se define la tecnología radio a implementar y el diseño del piloto dividido en módulos, en los que se especifica cada funcionalidad y la configuración necesaria para lograr la comunicación entre estos.

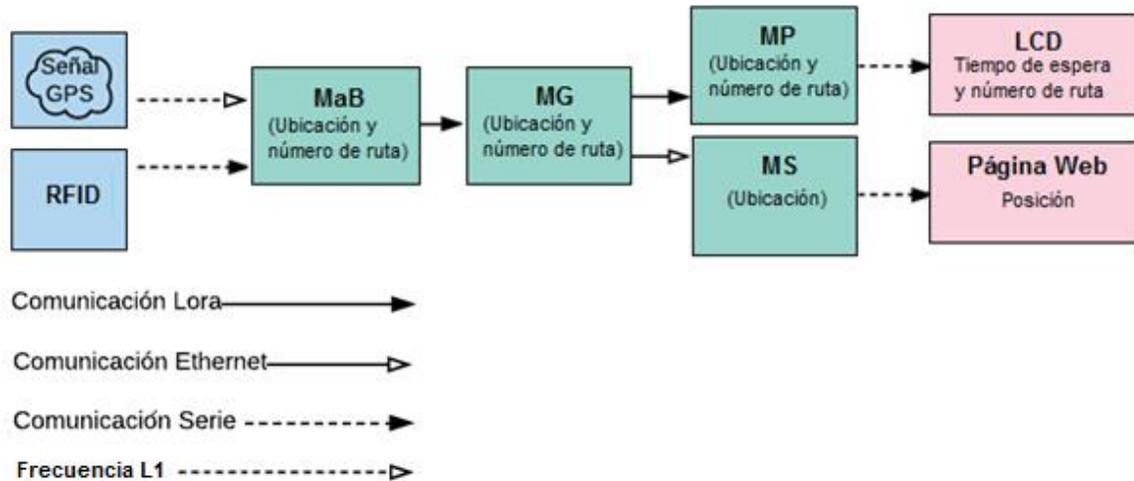
### 4.2.1. Tecnología radio para la transmisión de información en el piloto

El análisis realizado en el Capítulo 2 permitió definir la tecnología conveniente para el desarrollo del piloto; inicialmente se eligió el grupo de tecnologías LPWAN, teniendo en cuenta los parámetros establecidos, posterior a esto, se hizo un análisis más detallado de las principales tecnologías LPWAN, optando por la tecnología LoRa, la cual cumplió con la mayoría de las características requeridas.

### 4.2.2. Diseño del piloto de información de control de flota

En el diseño se especifican los cuatro módulos que en conjunto conforman el piloto de información: Modulo a Bordo (MaB), instalado en el bus; Modulo Gateway (MG), instalado en un lugar alto para proporcionar una cobertura apropiada; Modulo parada (MP), instalado en lugares definidos como las paradas del piloto y Modulo Servidor (MS), alojado en la nube. De igual manera se realiza el diseño de comunicación entre módulos, para permitir la recepción y transmisión de datos.

De forma general, el MaB es el encargado de recibir los datos de ubicación por medio de un GPS y obtener un número de ruta a través de un RFID, los envía al MG que actúa como repetidor, enviando los datos al MP y al MS, para ser procesados y desplegados en Pantallas de Cristal Líquido (LCD, *Liquid Crystal Display*) y página Web respectivamente. Los datos enviados al MP corresponden a la ubicación y número de ruta, mientras que al MS solo se envían datos de ubicación. En la *Figura 4.2* se observa el proceso descrito anteriormente, en un modelo esquemático del piloto.



**Figura 4.2 Modelo esquemático del piloto**

A continuación se detalla el diseño de cada módulo y la comunicación entre estos, en sus componentes hardware y software.

#### 4.2.2.1. Diseño de comunicación Lora

Con el diseño de comunicación LoRa se define la configuración adecuada que permita la transmisión y recepción de información entre módulos.

##### a) Software

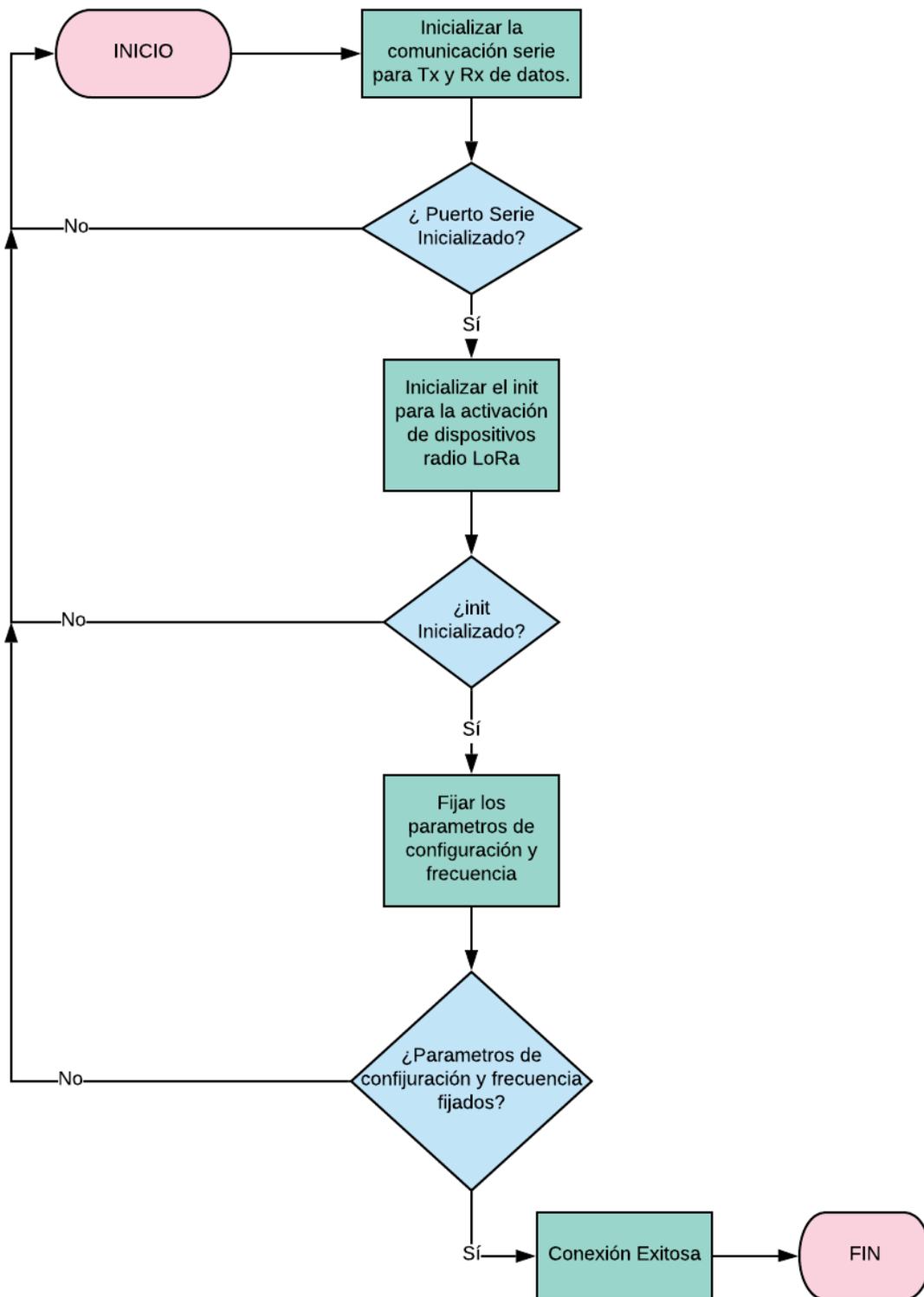
La comunicación LoRa se logra al fijar en los dispositivos radio, una frecuencia de operación y parámetros de configuración (BW, SF y CR), los cuales definen sobre la red, el rango de cobertura, la velocidad de transmisión de datos y la robustez frente a interferencias. La frecuencia de operación se selecciona teniendo en cuenta la regulación de frecuencias para la región en donde se opera y el soporte de frecuencias que tenga el dispositivo, se puede fijar una o dos frecuencias de acuerdo con la necesidad de cada módulo al transmitir y recibir datos. Los parámetros de configuración se eligen en base a cuatro configuraciones [62], determinadas por los desarrolladores de Dragino<sup>10</sup> como se observa en la *Tabla 4.1*, de las cuales se elige la configuración 1, ya que brinda una cobertura media, abarcando el escenario de pruebas del piloto y una velocidad de transmisión de datos aceptable para la recopilación de información necesaria.

<sup>10</sup> Dragino es OpenWrt, un software de código abierto que permite a los desarrolladores, modificar y diseñar libremente diferentes aplicaciones [81].

	Parámetros			Descripción
	BW	SF	CR	
<b>Configuración 1</b>	125 KHz	7	4/5	Medio + Medio alcance
<b>Configuración 2</b>	500 KHz	7	4/5	Rápido + Corto alcance
<b>Configuración 3</b>	32 KHz	9	4/8	Lento + Largo alcance
<b>Configuración 4</b>	125 KHz	12	4/8	Lento + Largo alcance

**Tabla 4.1 Configuraciones para dispositivos radio LoRa [62].**

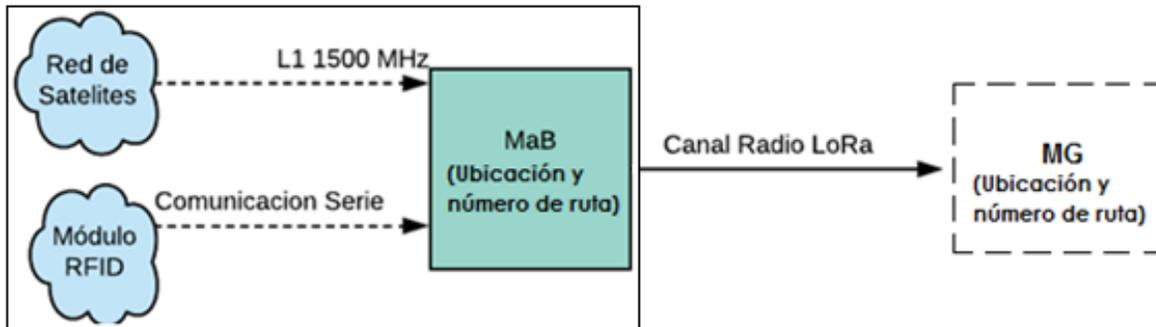
MaB, MG y MP requieren comunicación mediante LoRa, a diferencia del MS que requiere solo de conexión Ethernet para enviar los datos a la nube. Para establecer la comunicación LoRa es necesario inicializar un puerto serial para la transmisión y recepción de datos y un *init* que tiene la función de activar los dispositivos radio Lora, una vez inicializados los dispositivos y el puerto serie se procede a fijar los parámetros de configuración y frecuencia de operación. Este proceso se describe en el diagrama de flujo de la *Figura 4.3*.



**Figura 4.3 Diagrama de flujo para establecer una comunicación LoRa entre dispositivos radio.**

#### 4.2.2.2. Diseño del Módulo a Bordo

El MaB tiene la funcionalidad de recibir datos de ubicación por medio de un receptor GPS a una frecuencia de operación L1:1575 MHz, utilizando una antena de parche<sup>11</sup> (versión 15\*15\*4 mm), obtener el número de ruta del bus a través de un lector RFID y transmitir los datos recogidos al MG por un canal radio LoRa asignado. La *Figura 4.4* muestra el modelo esquemático del MaB.



**Figura 4.4 Modelo esquemático del MaB**

##### a) Hardware

Es necesario un transceptor LoRa de largo alcance que tenga incorporado un receptor GPS, una placa Arduino para el procesamiento de datos y un módulo de Identificación por Radiofrecuencia<sup>12</sup> (RFID, *Radio Frequency Identification*) compatible con Arduino. En base a lo mencionado anteriormente, costo y disponibilidad de equipos, se seleccionan los dispositivos: LoRa/GPS Shield v95, Arduino uno y un módulo RFID RC-522.

Características de dispositivos:

- LoRa/GPS Shield v95, es una placa de expansión de LoRa/GPS desarrollada por Dragino para usarse con Arduino, destinado a solucionar aplicaciones LoRa que requieran datos GPS. Se basa en el transceptor sx1276 /sx1278<sup>13</sup> y en la tecnología *AlwaysLocate* para GPS, que puede ajustar el adaptador de tiempo de encendido/ apagado para lograr el

<sup>11</sup> Es el tipo de antena más común para aplicaciones de GPS, son planas, generalmente tienen un cuerpo de cerámica y metal y están montadas sobre una placa de metal.

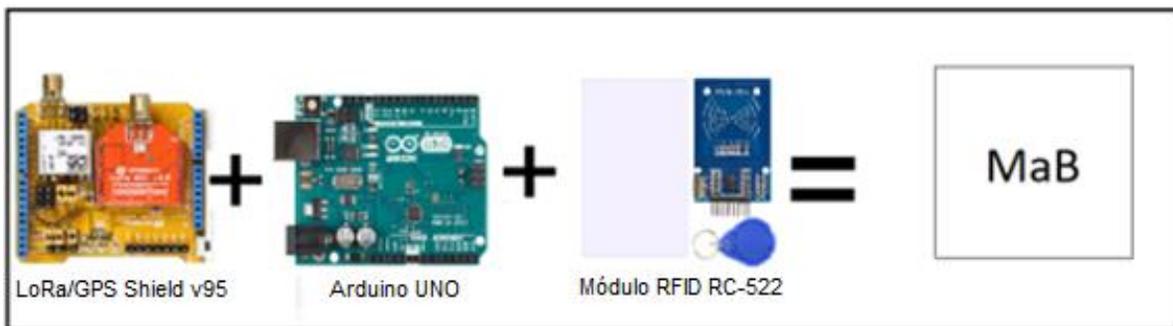
<sup>12</sup> Identificación por RadioFrecuencia (RFID, *Radio Frequency Identification*): Sistema de almacenamiento y recuperación de datos remotos que usa dispositivos denominados etiquetas o tarjetas RFID. Tiene el propósito de transmitir la identidad de un objeto mediante ondas radio.

<sup>13</sup> Transceptor SX1276 y Sx1278 dispositivo sintonizado a frecuencia de 433 MHz y 868/915 MHz.

equilibrio entre la precisión de posicionamiento y el consumo de energía según las condiciones ambientales y de movimiento [63].

- Arduino uno, es una placa de microcontrolador basada en el ATmega 328P y desarrollada por Arduino.cc<sup>14</sup>, de código abierto y de hardware y software flexible, equipada con un conjunto de pines de entrada/salida (analógico/digital) que pueden conectarse a varias tarjetas de expansión y otros circuitos; un cristal de cuarzo de 16 MHz, una conexión USB, un conector de alimentación, un encabezado de Programación Serial en Circuito de Corriente alterna<sup>15</sup> (ICSP, *In-Circuit Serial Programming*) y un botón de reinicio [64].
- Módulo RFID RC-522, es un lector de RFID que trabaja a 3.3v, puede controlarse usando el protocolo de comunicación SPI y es compatible con casi cualquier microcontrolador Arduino o tarjeta de desarrollo. Es utilizado para comunicación inalámbrica a 13.56 MHz, para escribir o leer datos de aplicaciones de bajo consumo de energía, bajo costo y tamaño reducido [65].

Se realiza el diseño esquemático, observado en la *Figura 4.5*, con los componentes hardware requeridos para conformar el MaB, posteriormente se establece el diseño circuital, en el cual se tienen en cuenta los pines disponibles del dispositivo LoRa/GPS Shield v95 montado sobre la placa Arduino uno, con el fin de evitar problemas de transmisión y recepción de datos [66].

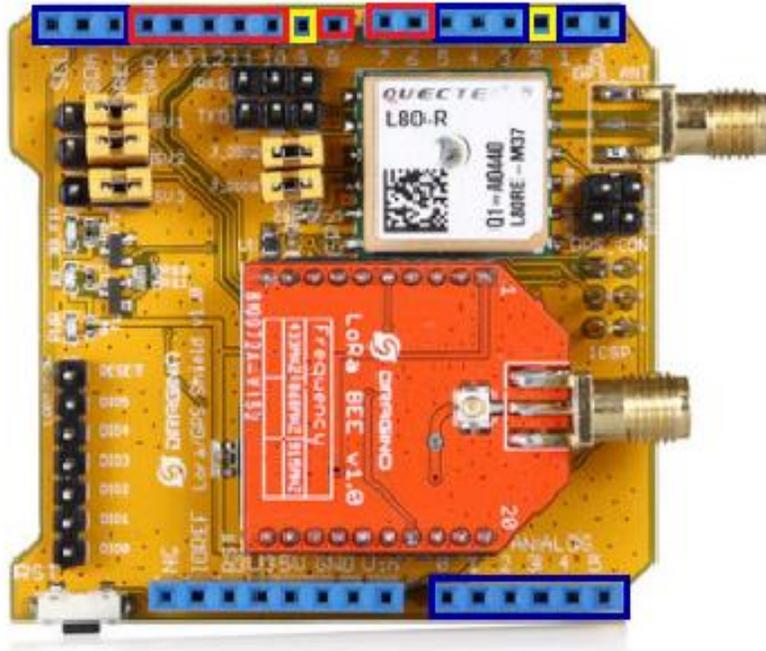


**Figura 4.5** Diseño esquemático del MaB. Por los autores

<sup>14</sup> Compañía encargada de fabricar una gama de herramientas de software, plataformas de hardware y documentación, para el desarrollo de aplicaciones en múltiples áreas. Arduino crea dispositivos de código abierto, basados en una placa conformada por un microcontrolador y un entorno de desarrollo, diseñados para facilitar la creación de aplicaciones [80].

<sup>15</sup> ICSP: Habilidad de algunos microcontroladores de ser programados, mientras están instalados en un sistema completo, en lugar de requerir que el chip sea programado antes de ser instalado dentro del sistema.

La *Figura 4.6* muestra los pines disponibles y no disponibles, entendiendo que los pines en color amarillo no deben usarse, ya que cuentan con una preconfiguración de fábrica, los pines en color rojo se usan mientras no se presente ningún conflicto de comunicación y los pines en color azul están libres para su uso.



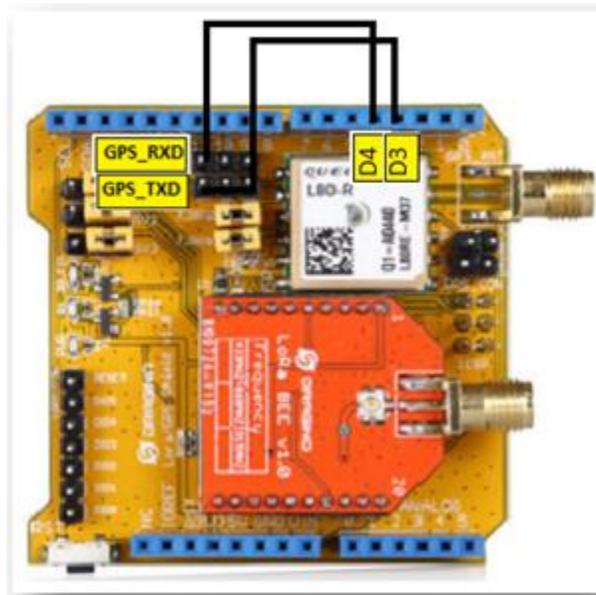
**Figura 4.6 Disponibilidad de pines LoRa/GPS Shield v9. Por los autores.**

La *Tabla 4.2* especifica la disponibilidad de pines, bajo la asignación de la *Figura 4.6*.

<b>Pines</b>	<b>Disponibilidad</b>
SCL	Disponible
SDA	Disponible
REF	Disponible
GND	Disponible con condición
D13	Disponible con condición
D12	Disponible con condición
D11	Disponible con condición
D10	Disponible con condición
D9	No disponible
D8	Disponible con condición
D7	Disponible con condición
D6	Disponible con condición
D5	Disponible
D4	Disponible
D3	Disponible
D2	No disponible
D1	Disponible
D0	Disponible
A0	Disponible
A1	Disponible
A2	Disponible
A3	Disponible
A4	Disponible
A5	Disponible

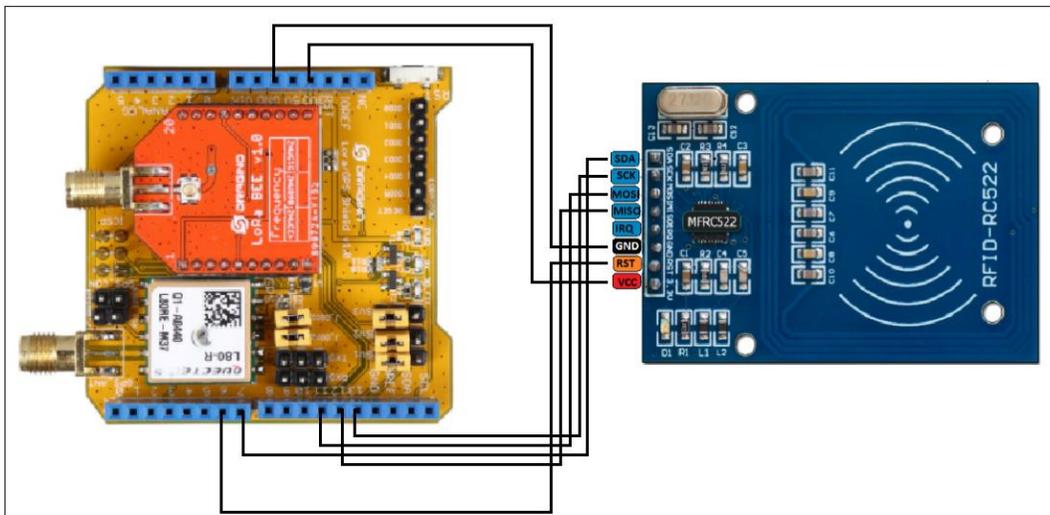
**Tabla 4.2 Disponibilidad de pines LoRa/GPS Shield v95. Por los autores.**

En la *Figura 4.7* se observa el diseño circuital entre el GPS y el Arduino uno, de esta manera, el GPS recibirá los datos y la tarjeta Arduino podrá procesarlos.



**Figura 4.7** Diseño circuital del GPS y Arduino uno. Por los autores.

Para obtener datos de número de ruta, es necesaria la unión del módulo RFID RC-522 con el montaje del LoRa/GPS Shield v95 sobre el Arduino uno, como se muestra en el diagrama circuital de la *Figura 4.8*, basándose en la asignación de pines detallada en la *Tabla 4.3*.



**Figura 4.8** Diagrama circuital LoRa/GPS Shield montado sobre Arduino con el módulo RFID RC-522. Por los autores.

Asignación de pines entre Arduino uno y módulo RFID RC-522	
Pines Arduino uno	Pines módulo RFID RC-522
3.3 Voltios	3.3 V
D6	RST
GND	GND
No conectado	IRQ
D12	MISO
D11	MOSI
D13	SCK
D7	SDA

**Tabla 4.3 Asignación de pines entre Arduino uno y módulo RFID RC-522. Por los autores.**

b) Software

El MaB debe establecer la comunicación radio para recibir la ubicación actual del bus en coordenadas geográficas<sup>16</sup> a través del receptor GPS y el número de la ruta por medio de un módulo RFID; estos datos son enviados al MG por un canal radio LoRa 1, configurado en una frecuencia de 868 MHz.

Para determinar el número de ruta del bus, se compara el código de la tarjeta de identificación dispuesta sobre el inicio de cada ruta<sup>17</sup>, con las claves de identificación establecidas en el código (se generan dos claves de identificación, las cuales asignan al bus un número de ruta igual a 1 ó 2), si coinciden se fija el número de ruta que le corresponde a cada clave de identificación.

El número de la ruta del bus además de ser información brindada al usuario sirve como identificador para determinar si la información corresponde a cada parada, ya que las paradas 1 y 2 escuchan datos de un bus con número de ruta igual a 1 y la parada 3 datos provenientes de un bus con número de ruta igual a 2. Este proceso se observa en el diagrama de flujo de la *Figura 4.8*.

<sup>16</sup>Coordenadas geográficas: sistema que referencia cualquier punto de la superficie terrestre y que utiliza para ello dos coordenadas angulares, latitud (norte o sur) y longitud (este u oeste).

<sup>17</sup> Se establecen dos rutas (1 y 2), para cada sentido de la carretera, con el fin de evitar que las paradas procesen información que no les corresponde.

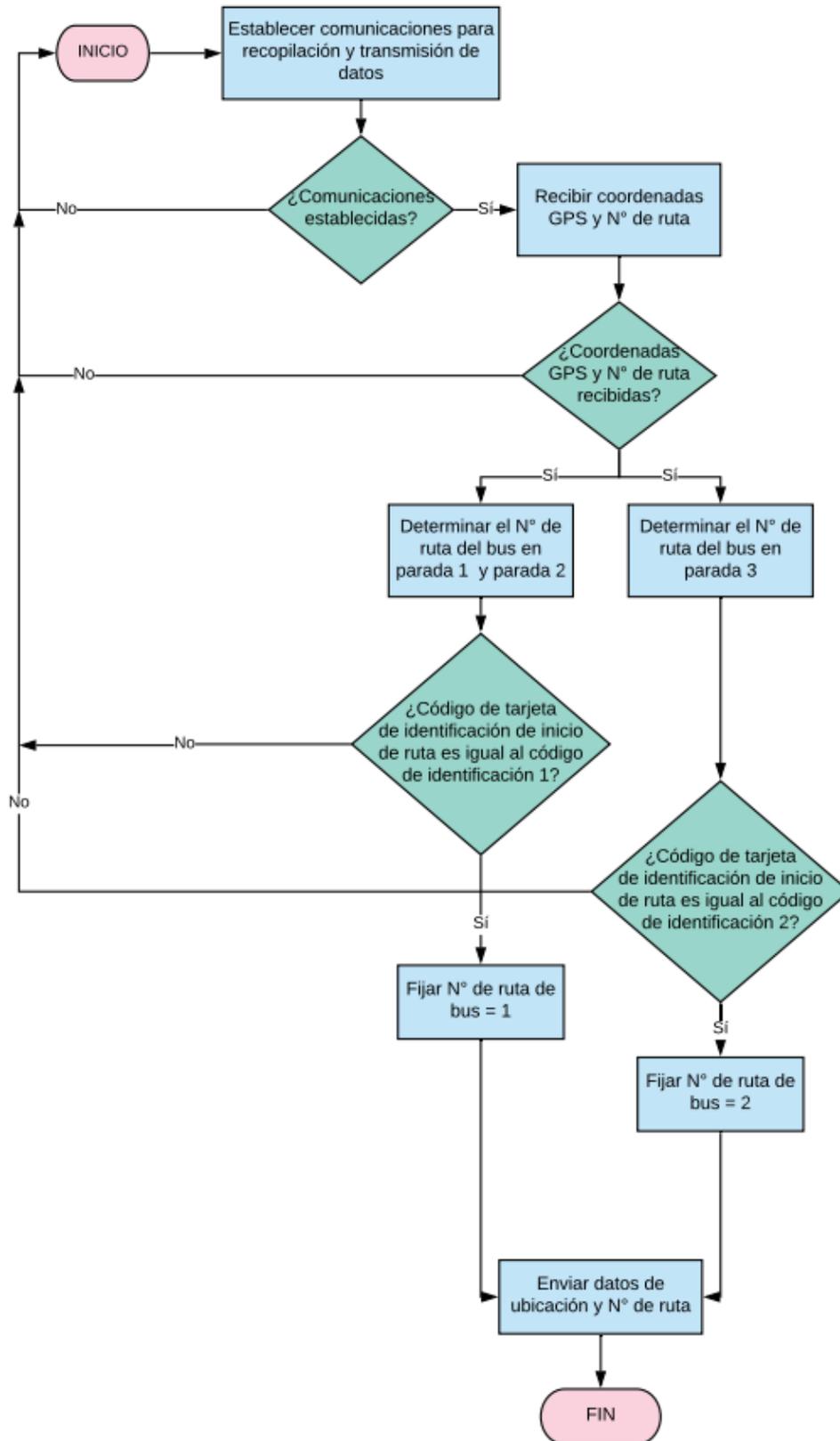
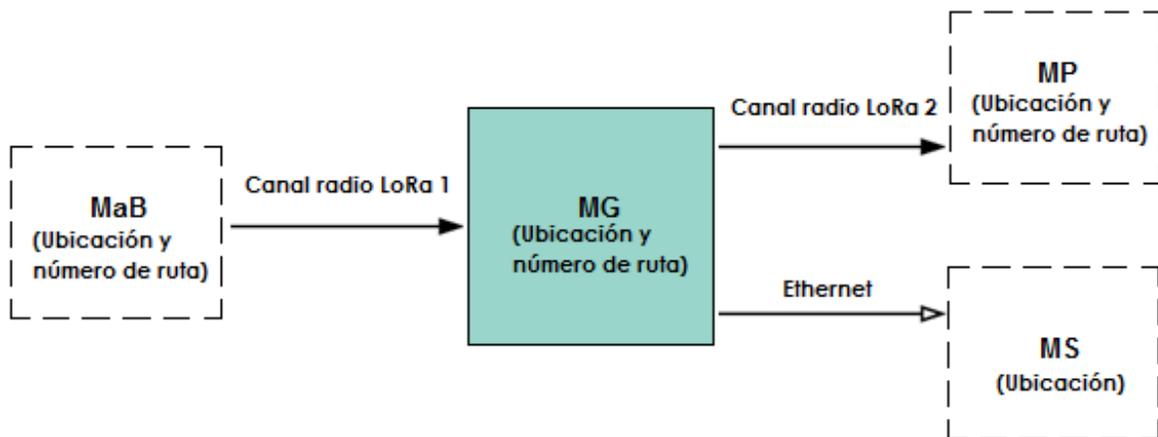


Figura 4.9 Diagrama de flujo para el MaB. Por los autores.

#### 4.2.2.3. Diseño Módulo Gateway

El MG se encarga de recibir la información proveniente del MaB mediante el canal radio LoRa 1 y transmitirla a los MP a través de un canal radio LoRa 2, con el fin de evitar posibles interferencias. Al mismo tiempo el MG debe enviar datos de ubicación al MS mediante comunicación Ethernet. La funcionalidad del MG es básica ya que no realiza un procesamiento de datos complejo, se enfoca en recibir y transmitir la misma información, sin embargo, el uso de este módulo es clave para proporcionar la cobertura adecuada al piloto. En la *Figura 4.10* se observa la funcionalidad del MG, mediante el modelo esquemático.



*Figura 4.10 Modelo esquemático del MG*

##### a) Hardware

El MG requiere de un dispositivo LoRa capaz de interconectar redes con protocolos y arquitecturas diferentes en todos los niveles de comunicación, ya que además de establecer su propia red, debe conectar la red inalámbrica LoRa a una red IP a través de Ethernet. Además se hace necesario el uso de un dispositivo que trabaje en exteriores y que cubra un área amplia de cobertura. Por lo dicho anteriormente, se elige el dispositivo Gateway LoRa OLG01.

Características del dispositivo:

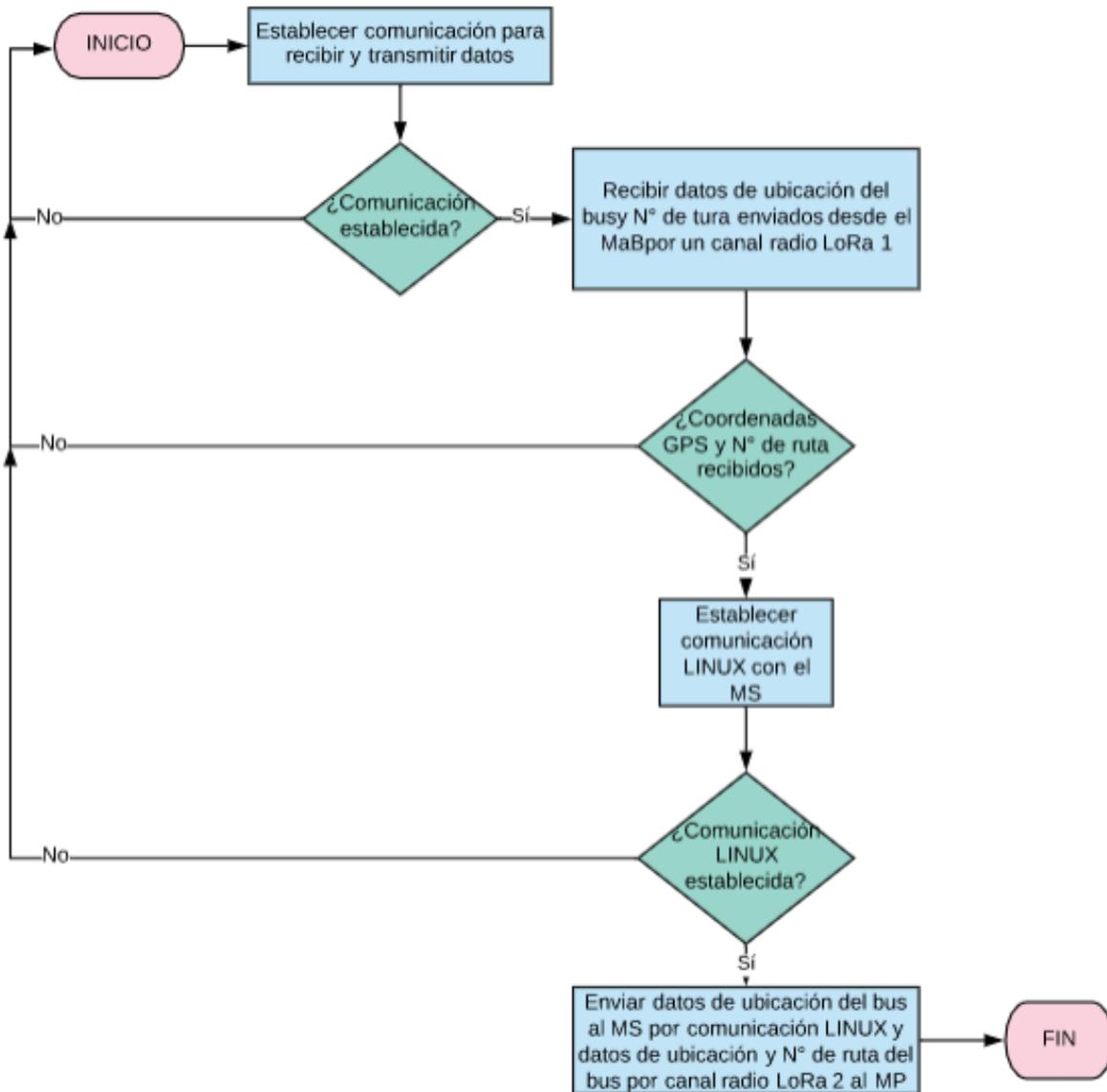
- Gateway LoRa OLG01, es una puerta de enlace LoRa de código abierto, fabricada por Dragino. Permite conectar la red inalámbrica LoRa a una red IP a través de WiFi, Ethernet, 3G o 4G celular. Interiormente cuenta con un microcontrolador ATmega328p, encargado de generar las comunicaciones

radio; posee dos puertos RJ45, utilizados para realizar la conexión a internet y la conexión con el computador de control, además sirve como vía para el suministro de corriente mediante del dispositivo de Alimentación a través de Ethernet (PoE, *Power over Ethernet*) [67].

El MG no necesita de un diseño circuital, ya que la Gateway LoRa OLG01 cuenta con un manual de usuario, en el cual se detallan los pasos a seguir para dar funcionamiento al dispositivo, por lo tanto, en este proyecto, el MG solo requiere desarrollo software, alimentación e interconexión con un computador y un punto de red, procesos que se especifican más adelante.

#### b) Software

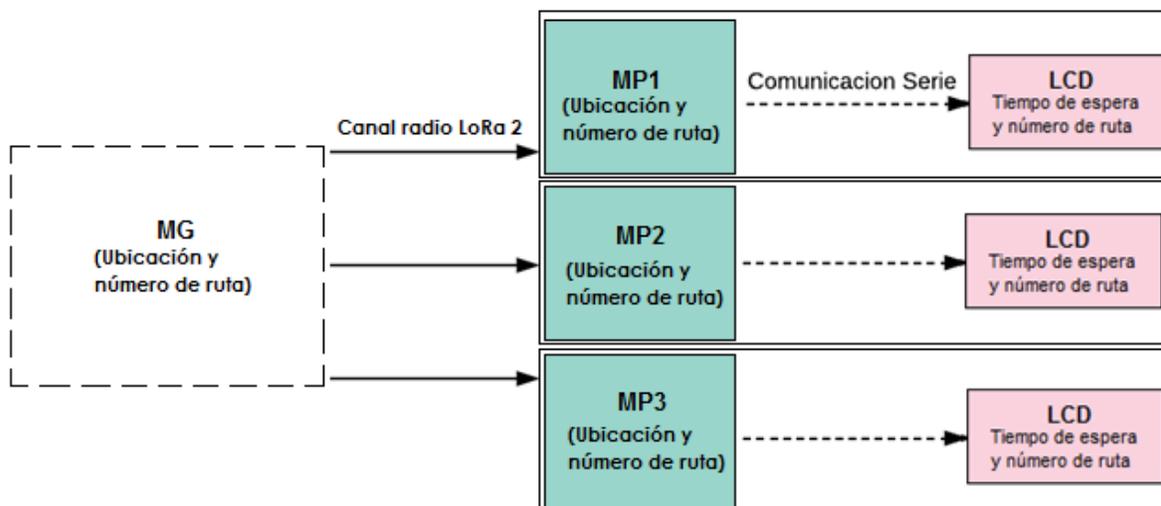
En el diseño software del MG se debe establecer comunicaciones radio de recepción y transmisión de datos, haciendo uso de los dos canales radio LoRa disponibles. La información proveniente del MaB se recibe mediante el canal 1, configurado en la frecuencia de 868 MHz y la transmite al MP por el canal 2, configurado en 915 MHz. La información de ubicación enviada al MS mediante Ethernet se realiza a través de comandos Linux. En la *Figura 4.11* se observa el diagrama de flujo que corresponde a este proceso.



**Figura 4.11 Diagrama de flujo para el MG. Por los autores**

#### 4.2.2.4. Diseño Modulo Parada

La funcionalidad del MP es más compleja, el MP recibe la información proveniente del MG, mediante el canal radio LoRa 2, posteriormente procesa los datos recibidos para determinar la información del tiempo de espera y desplegarla en pantalla. El diseño del piloto cuenta con 3 MP, en los cuales se realiza el mismo procedimiento pero se designan para rutas diferentes, visualizando en pantalla la información correspondiente a cada parada.



**Figura 4.12 Modelo esquemático del MP**

a) Hardware

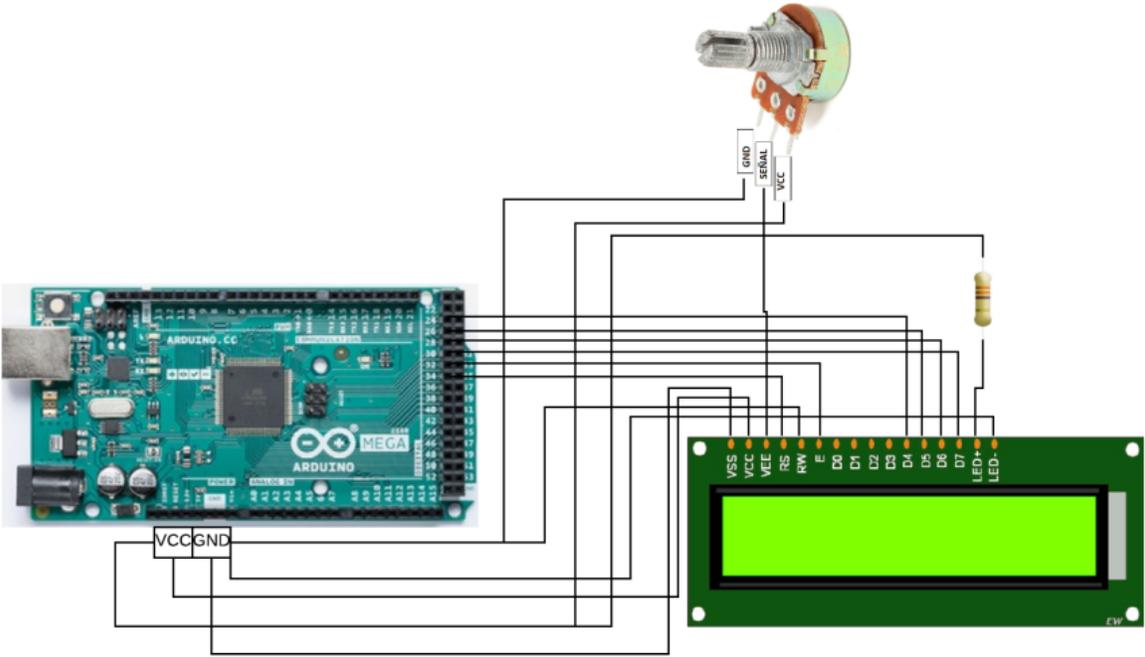
El MP requiere de tres dispositivos: un transceptor LoRa, que permita la recepción de datos provenientes del MG; una placa Arduino con mayor procesamiento, para leer, calcular y desplegar la información; una pantalla compatible con Arduino, adecuada para la visualización de los datos correspondientes al tiempo de espera y número de ruta. Se eligen los dispositivos LoRa Shield v95, Arduino Mega 2560 y una LCD 1602 para conformar el MP.

Características de los dispositivos:

- LoRa Shield v95, es una placa fabricada por Dragino, la cual permite comunicaciones a través de la tecnología LoRa y es compatible con Arduino. A diferencia de la placa LoRa/GPS Shield v95, esta no cuenta con un sistema incorporado de GPS y opera en las bandas 433 Mhz / 868 Mhz / 915 Mhz según una preconfiguración en fábrica. El LoRa Shield v95 está basado en el transceptor SX1276 / SX1278, permite enviar datos y alcanzar amplias áreas de cobertura a bajas tasas de datos. Proporciona alta inmunidad a interferencias minimizando el consumo de corriente [68].
- El Arduino Mega 2560, es una tarjeta de desarrollo de código abierto, desarrollada por Arduino. Está basada en un microcontrolador ATmega8U2 y posee pines de entradas y salidas (E/S), analógicas y digitales de las cuales 14 pueden ser utilizadas como salidas de Modulación por Ancho de Pulsos (PWM, *pulse width modulation*) y 16 como entradas analógicas, 4 UARTs (puertos serial por hardware), cristal oscilador de 16MHz, conexión USB, conector ICSP y botón de reset [69].



El circuito diseñado para establecer la conexión entre el dispositivo LoRa Shield v95 montado sobre el Arduino Mega 2560 y la LCD 1602, se representa en la *Figura 4.15*.



**Figura 4.15 Diagrama circuital LoRa Shield v95 montado sobre Arduino Mega 2560 y la LCD 1602. Por los autores.**

La *Tabla 4.4* presenta la asignación de pines entre el Arduino Mega 2560 y la LCD 1602.

Asignación de pines entre Arduino Mega 2560, LCD 1602 y Potenciómetro	
Pines Arduino Mega 2560	Pines LCD 1602
GND	VSS
VCC	VCC
D34	RS
GND	RW
D32	E
No conectado	D0
No conectado	D1
No conectado	D2
No conectado	D3
D24	D4
D26	D5
D28	D6
D30	D7
Pines Arduino Mega	Potenciómetro
VCC	PIN VCC
GND	PIN GND
Pines LCD 1602	Potenciómetro
VEE	PIN SEÑAL

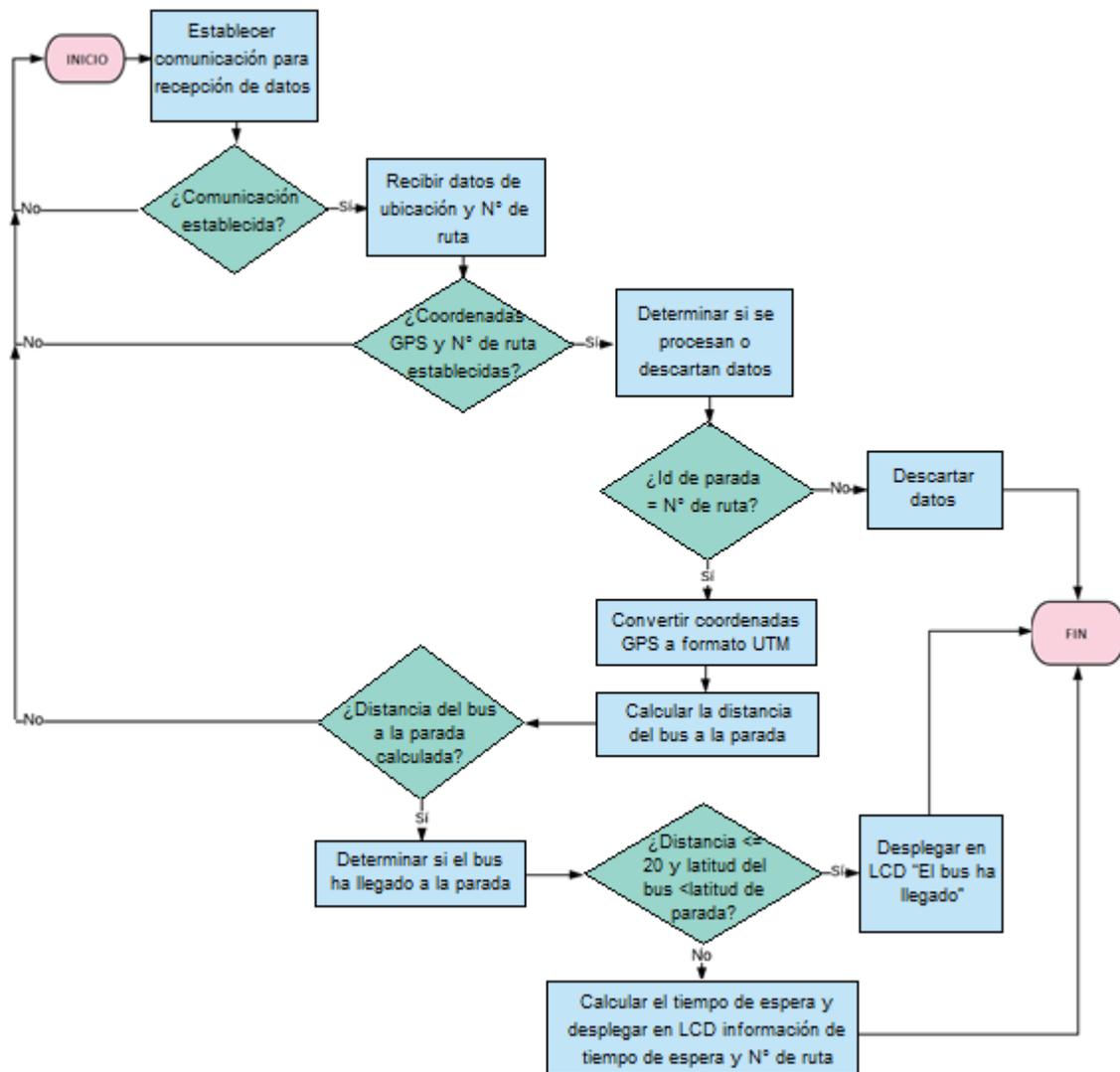
**Tabla 4.4 Asignación de pines entre Arduino MEGA, LCD y Potenciómetro. Por los autores.**

#### b) Software

El código a desarrollar debe permitir la recepción de datos correspondientes a la ubicación del bus y el número de ruta mediante el canal radio LoRa 2, enviados desde el MG. En primer lugar, se debe comparar si del MG se recibe el número de ruta que le corresponde, si es así, procede a realizar cálculos de distancia a partir de las coordenadas geográficas recibidas, si el número de ruta no corresponde a la parada, los datos son descartados.

Identificando que los datos corresponden a la parada, se hace una conversión de las variables correspondientes a latitud y longitud en formato de coordenadas

geográficas a formato UTM<sup>18</sup>, con el propósito de hacer operaciones matemáticas para determinar la distancia del bus a la parada. Se desarrolla un procedimiento lógico, en el cual se compara si la distancia del bus a la parada es menor o igual a 20 metros y si la latitud del bus es menor a la latitud de la parada, para desplegar en pantalla que el bus ha llegado; de lo contrario se hacen operaciones para encontrar el tiempo de espera con el uso de la formula  $d = v * t$ , esto, con el fin de darle información al usuario de tiempo de espera y número de ruta. Este procedimiento se observa en el diagrama de flujo de la *Figura 4.16*.

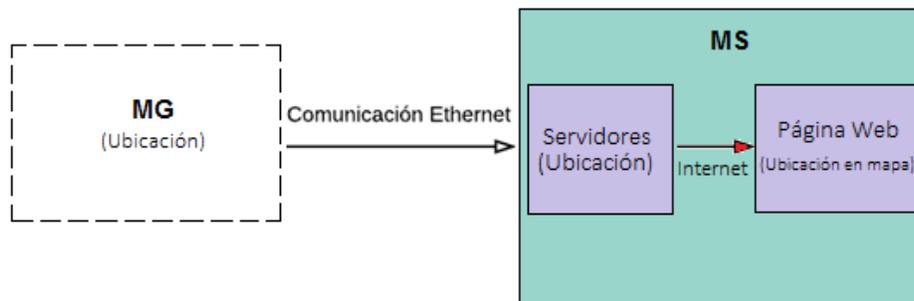


**Figura 4. 16 Diagrama de flujo general para el MP. Por los autores.**

<sup>18</sup> El Sistema de Coordenadas Universal Transversal de Mercator (UTM, *Universal Transverse Mercator*) está basado en la proyección geográfica transversa de Mercator, pero en vez de hacerla tangente al Ecuador, se la hace tangente a un meridiano. Las magnitudes en el sistema UTM se expresan en metros al nivel del mar.

#### 4.2.2.5. Diseño del Módulo Servidor

El MS es el encargado de proyectar la posición del bus en tiempo real sobre mapas mediante una página Web. El MG envía los datos de ubicación del bus al MS a través de Ethernet, los cuales son almacenados en servidores y extraídos mediante código HTML para ser visualizados en un mapa. El modelo esquemático del MS se observa en la *Figura 4.17*.



**Figura 4.17 Modelo esquemático del MS**

##### a) Hardware

El MS no requiere un diseño hardware, debido a que hace uso de la infraestructura brindada por los servidores seleccionados.

##### b) Software

El MS debe recopilar, gestionar la información y proporcionar el mapa necesario para mostrar al usuario la ubicación en tiempo real del bus sobre un escenario establecido. Por lo anterior se requiere un servidor Web gratuito para IoT compatible con Arduino y un servidor de aplicaciones de mapas en la Web que permita la gestión de mapas. Se selecciona el servidor ThingSpeak y el servidor Google Maps, que en conjunto con una página Web conforman el MS.

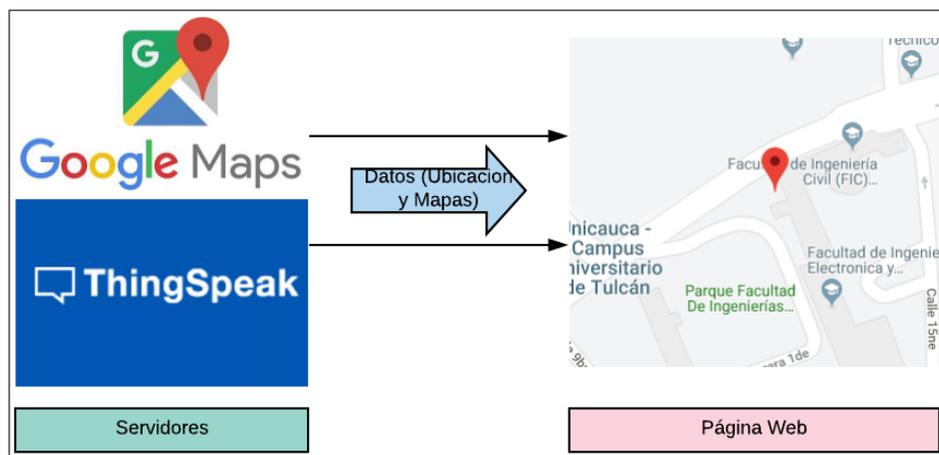
Características de los servidores:

- Servidor ThingSpeak, es un servidor Web gratuito de código abierto, lanzado originalmente por *ioBridge* en 2010 para aplicaciones de IoT, para recopilar, almacenar datos de sensores en la nube mediante el Protocolo de

Transferencia de Hipertexto<sup>19</sup> (HTTP, *HyperText Transfer Protocol*), a través de internet o mediante una red de área local. ThingSpeak proporciona aplicaciones que le permiten analizar y visualizar sus datos en Matlab y luego actuar sobre ellos. Es compatible con *Arduino*, *Raspberry Pi*, *BeagleBone Black* y otros tipo de hardware [71].

- Servidor Google Maps, es un servidor de aplicaciones de mapas en la web que pertenece a *Alphabet Inc.* Ofrece imágenes de mapas desplazables, así como fotografías por satélite del mundo e incluso la ruta entre diferentes ubicaciones o imágenes a pie de calle, condiciones de tráfico en tiempo real y un calculador de rutas a pie, en carro, bicicleta y transporte público, y un navegador GPS

Se debe establecer la comunicación del servidor ThingSpeak y el Servidor de Google Maps a una página Web, en donde se obtienen los datos del servidor ThingSpeak y del servidor de Google Maps, los cuales son procesados utilizando HTML<sup>20</sup> y la API<sup>21</sup> de JavaScript de Google<sup>22</sup>. Los datos requeridos en la página Web son la ubicación del bus en coordenadas de latitud y longitud y mapas satelitales. La *Figura 4.18* muestra el diseño esquemático de la comunicación.



**Figura 4.18** Diseño esquemático de la comunicación entre servidores y la página Web.

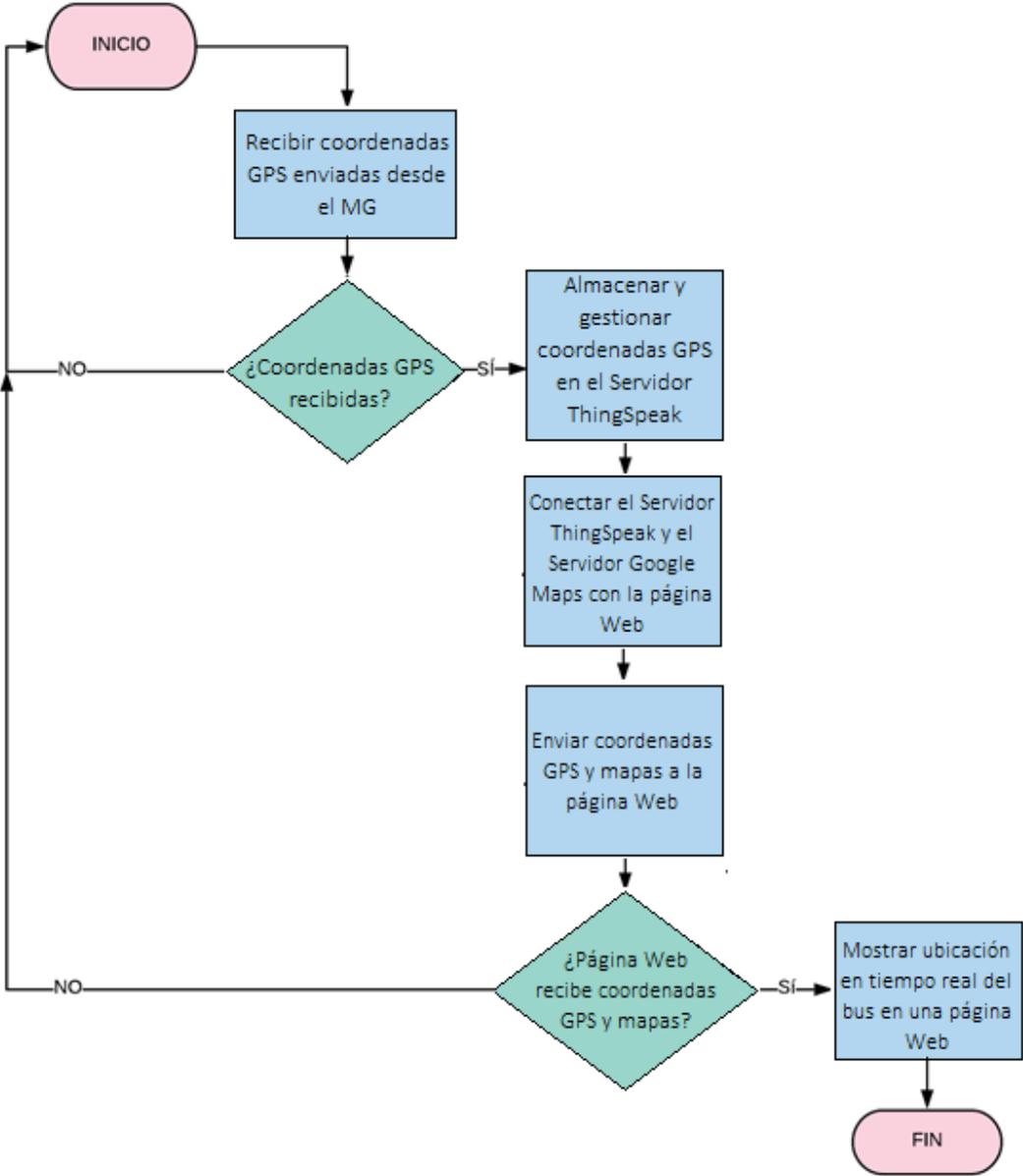
<sup>19</sup> HTTP, es el protocolo de comunicación que permite las transferencias de información en la World Wide Web.

<sup>20</sup> Lenguaje de Marcas de Hipertexto (HTML, *HyperText Markup Language*), hace referencia al lenguaje de marcado para la elaboración de páginas Web, define una estructura básica y un código denominado HTML, para la definición de contenido de una página Web, como texto, imágenes, videos, juegos, entre otros.

<sup>21</sup> Interfaz de Programación de Aplicaciones (API, *Application Programming Interface*), es un conjunto de funciones y procedimientos que cumplen una o muchas funciones con el fin de ser utilizadas por otro software.

<sup>22</sup> API de JavaScript de Google, permite personalizar mapas con su propio contenido e imágenes para mostrarlos en páginas Web y dispositivos móviles. Presenta cuatro tipos de mapas básicos hoja de ruta, satélite, híbrido y terreno; que se pueden modificar utilizando capas y estilos, controles y evento, y diversos servicios y bibliotecas.

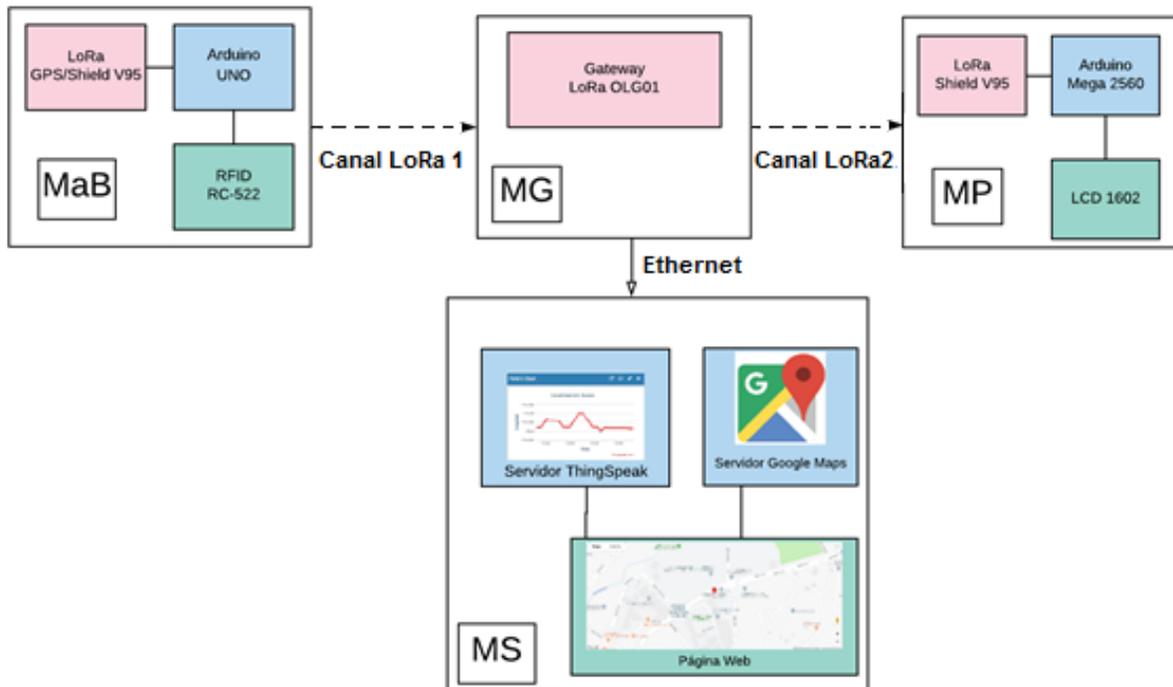
Los datos proporcionados por el MG se almacenan en el servidor ThingSpeak por medio de una conexión IP vía Ethernet y una API REST<sup>23</sup>, una vez almacenados en el servidor ThingSpeak, deben ser gestionados junto con los datos de Mapas proporcionados por el servidor de Google Maps y se envían a la página Web para finalmente ser mostrados en un mapa. Este proceso se explica mediante el diagrama de flujo en la *Figura. 4.19*.



**Figura 4.19 Diagrama de flujo MS. Por los autores.**

<sup>23</sup> Transferencia de Estado Representacional (*REST, Representational State Transfer*), es un estilo de arquitectura software para sistemas hipermedia distribuidos como la *World Wide Web*.

Se finaliza la etapa de diseño definiendo los procesos de comunicación y desarrollo de cada módulo. Concibiendo la comunicación entre los módulos por medio de la tecnología LoRa, haciendo uso de dos canales radio LoRa y la comunicación hacia el servidor a través de conexión IP. Además se especifican los diagramas de flujo correspondientes a los procesos que se deben realizar. Finalmente se obtiene el diseño general del piloto, presente en la *Figura 4.20*.



**Figura 4.20** Diseño general del piloto de información de control de flota. Por los autores

El diagrama de flujo de la *Figura 4.21* determina la lógica requerida para que el piloto proporcione la información al usuario de tiempo de espera del bus a la parada, el número de la ruta y la ubicación en tiempo real del bus.

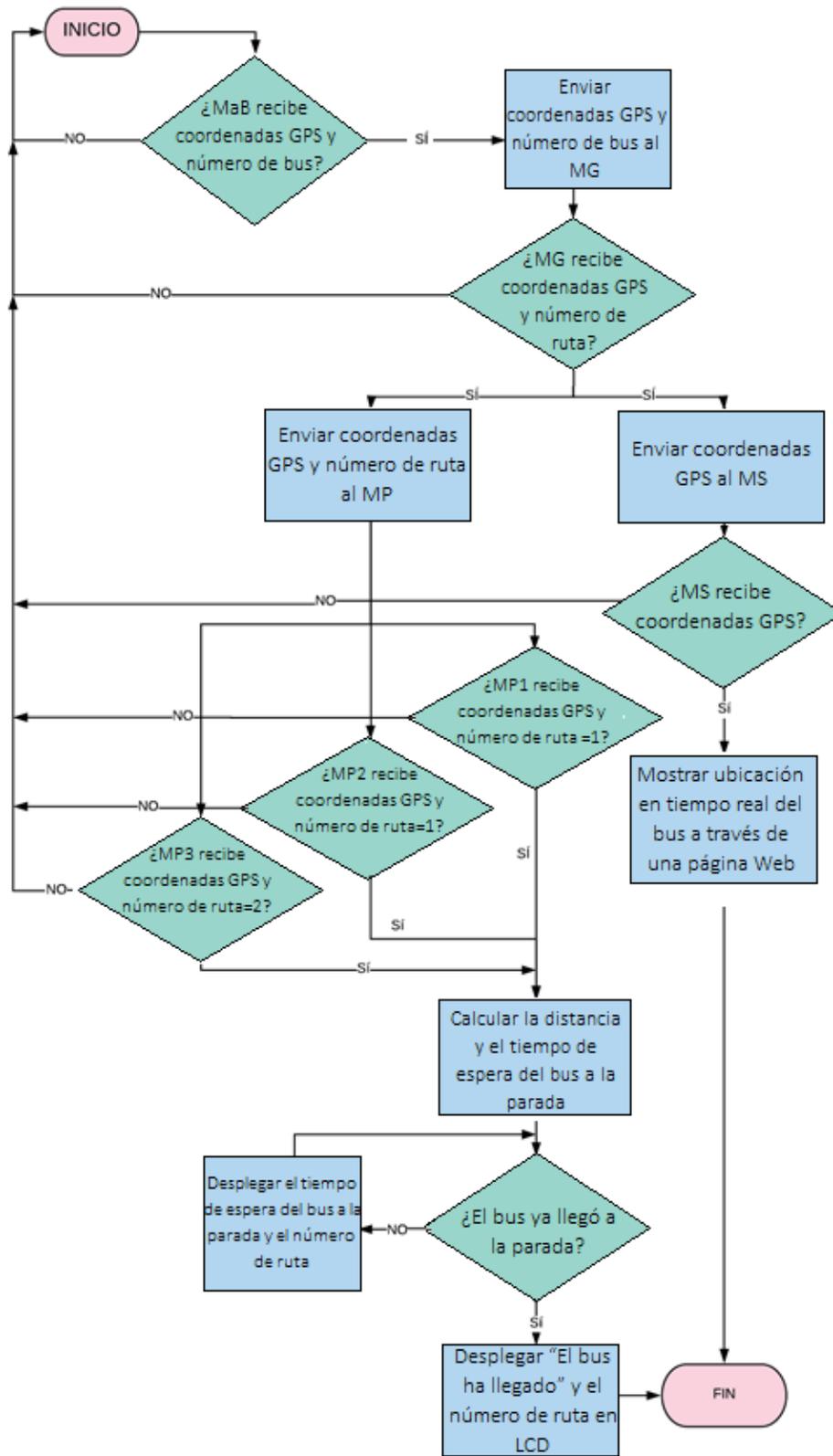


Figura 4.21 Diagrama de flujo del piloto de información al usuario. Por los autores

### 4.3. IMPLEMENTACIÓN

La implementación del presente trabajo de grado, se lleva a cabo por medio de los componentes hardware, software y el proceso de comunicación entre módulos; teniendo en cuenta los dispositivos seleccionados y la lógica correspondiente.

Para la implementación software, se detallan cuatro librerías utilizadas durante el desarrollo del piloto, las cuales simplifican y facilitan el código creado.

- Librería Radiohead: Es una biblioteca orientada a objetos para enviar y recibir mensajes en paquetes a través de una variedad de radios y microprocesadores integrados, compatible con diferentes entornos de desarrollo como Arduino. Radiohead desarrolla diferentes *drivers* para proporcionar acceso a los datos o al transporte de la radio; en este caso se hace uso del *driver* RH\_RF95, específico para el transceptor sx1276 /sx1278 usado en el desarrollo de los módulos diseñados anteriormente [72].
- Librería TinyGPS: Es una librería para analizar los flujos de datos en formato NMEA 0183<sup>24</sup> proporcionados por los módulos GPS, los datos en este formato son de poco entendimiento por lo cual la librería brinda métodos compactos y fáciles de usar para extraer la posición, fecha, hora, altitud, velocidad y rumbo de los dispositivos GPS [73].
- Librería RFID-MASTER: Es la librería encargada de controlar la comunicación entre el Arduino y el módulo RFID RC-522, haciendo uso del *driver* MFRC522.
- Librería LiquidCrystal: Es una librería que permite controlar las pantallas LCD basadas en el controlador Hitachi HD44780 o uno compatible, que se encuentra en la mayoría de las LCD basadas en texto.

#### 4.3.1. Implementación de la comunicación LoRa y conexiones del MG

##### a) Comunicación LoRa entre módulos

Al configurar los parámetros y la frecuencia de operación, se desarrolla el software necesario que permita establecer la comunicación LoRa entre los dispositivos. Se

---

<sup>24</sup> NMEA 0183: Es un protocolo que permite la comunicación entre dispositivos electrónicos marinos y receptores GPS, controlados por la organización estadounidense *National Marine Electronics Association*.

hace uso de las librerías `SoftwareSerial` y `Radiohead`, utilizando su *driver* `RH_RF95`, el cual permite recibir y transmitir datagramas a través de dispositivos LoRa. El pseudocódigo que ejecuta este proceso se muestra a continuación:

```
Inicio Setup
Inicializar Serial ← 9600
  Mientras ( Serial disponible)
    Sí (init disponible) Entonces
      Mientras (1)
        Fin Mientras
    Fin sí
  Fin Mientras
Fijar Frecuencia ← 868.0
Fijar Factor de Ensanchamiento ← 7
Fijar Ancho de Banda ← 125
Fijar Tasa de Codificación ← 4/5
Fijar Potencia de Transmisión ← 20
Escribir ("Ready to Send")
Fin Setup
```

*Figura 4. 22 Pseudocódigo para establecer comunicación LoRa entre dispositivos radio.*

La transmisión de paquetes entre dispositivos radio LoRa se logra al fijar una frecuencia de trasmisión de datos continuando con la ejecución de la función envío de paquetes; una vez enviado el paquete se espera un tiempo determinado por la función *waitPacketSent*, que finalmente envía una señal informando que el canal está disponible para transmitir nuevos datos. El pseudocódigo de la transmisión de paquetes se muestra en la *Figura 4.23*:

```

Inicio Loop

Fijar Frecuencia ← 915.0

Enviar (datasend con tamaño de (datasend))

Esperar Paquete a enviar

Esperar (4000)

Fin Loop

```

**Figura 4. 23 Pseudocódigo de la trasmisión de paquetes en LoRa.**

Los dispositivo radio LoRa para recibir paquetes exitosamente, debe verificar si existe una conexión LoRa y activar el receptor haciendo uso de la función *rf95.recv(indatabuf, &len)*, una vez recibe los datos, los almacena en una variable llamada *indatabuf* y configura un tamaño *len* correspondiente a la longitud máxima de mensaje que soporta el controlador. El pseudocódigo se muestra a continuación:

```

Procedimiento receivepacket

indatabuf : uint8_t
indatabuf ← 251 bytes
len : uint8_t
len ← tamaño de indatabuf

    si (el tiempo de espera está dentro de 3 segundos) Entonces
        Si (recibir indatabuf de tamaño len) Entonces
            Desarrollo lógico
                Si-No
                    Escribir ("Recepción de datos Errónea")
                Fin Si
            Si-No
                Escribir ("No hay respuesta del Servidor LoRa, ¿Está disponible?")
            Fin Si
        Fin Procedimiento

```

**Figura 4.24 Pseudocódigo de la recepción de paquetes en LoRa**

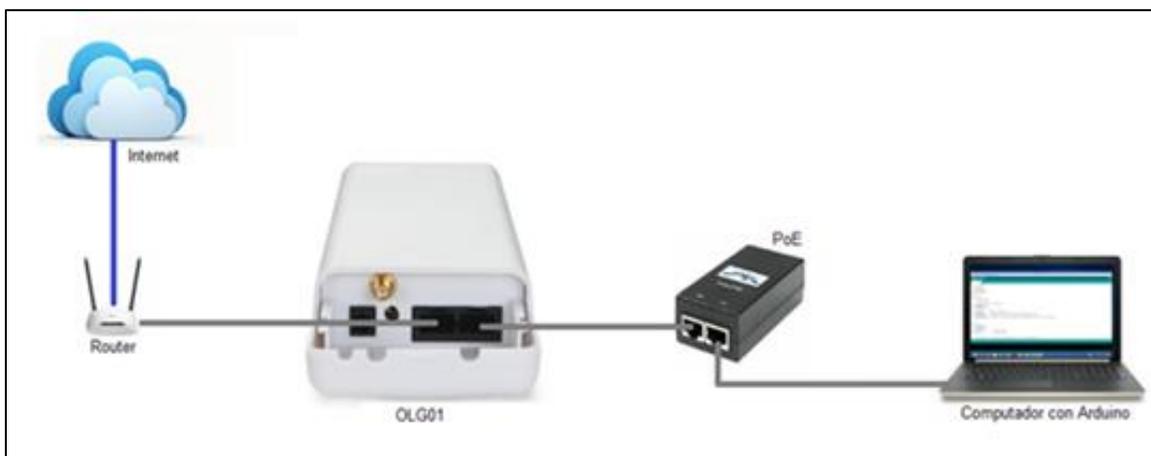
Todos los mensajes recibidos y enviados por el controlador RF\_RH95 se ajustan al formato de paquete mostrado a continuación:

- 8 símbolos de preámbulo.
- Encabezado explícito con encabezado CRC (Manejado internamente por la radio).
- 4 octetos de encabezado.
- 0 - 251 octetos de datos.
- CRC (Manejado internamente por la radio).

#### b) Conexiones del MG

Para que el MG establezca su propia red LoRa y tenga acceso a Internet, es necesario conectar mediante Ethernet los puertos LAN y WAN de la Gateway hacia un computador y un punto de acceso respectivamente, para ello se hace uso de un dispositivo PoE, el cual también sirve como fuente de alimentación.

Este proceso consiste en conectar el puerto WAN de la Gateway hacia un Router, el puerto LAN hacia el puerto POE de un adaptador POE, el cual se conecta a una fuente de alimentación DC y el puerto Ethernet, del mismo adaptador al puerto Ethernet del computador; la *Figura 4.25* muestra lo descrito anteriormente.



**Figura 4.25 Conexión Gateway LoRa OLG01. Por los autores.**

### 4.3.2. Implementación Software

La implementación software desarrolla la lógica de cada módulo, lo cual se logra haciendo uso del entorno de programación de Arduino IDE y de librerías.

#### 4.3.2.1. Implementación software Módulo a Bordo

Para determinar el número de ruta del bus, se hace necesario el uso de la librería MFRC522 para el control de transmisión y recepción de datos del módulo RFID. Como se muestra en el pseudocódigo de la *Figura 4.26*.

```
Iniciar Proceso  
ActualUID[4]: byte  
Usuario1[2] ← {0x50, 0xBA, 0x81, 0x7A}  
Usuario2[2] ← {0x96, 0x33, 0x98, 0xBB}  
Iniciar Loop  
  Sí (Hay tarjetas de identificación disponibles)  
    Sí (Tarjeta de identificación ha sido leída)  
      Escribir (Código de identificación de tarjeta)  
      ActualUID ← Código de identificación de tarjeta  
      Sí (ActualUID=Usuario1)  
        IdBus ← 1  
        Escribir ("Acceso concedido")  
        Escribir (IdBus)  
      Fin Sí  
    Si-No  
      Sí (ActualUID=Usuario2)  
        IdBus ← 2  
        Escribir ("Acceso concedido")  
        Escribir (IdBus)  
      Fin Sí  
    Fin Sí  
  Fin Loop  
Fin Proceso
```

*Figura 4.26 Pseudocódigo para la obtención del número de ruta*

Se usa la librería TinyGPS en conjunto de su función `gps.f_get_position(&flat, &flon, &age)` para extraer los datos del GPS en formato NMEA, a un formato de fácil manipulación correspondiente a entero flotante, los datos que se extraen corresponden a latitud, longitud y hora. Una vez recolectados los datos en un formato adecuado se concatenan con el número de ruta para ser transmitidos por un canal radio LoRa hacia el MG. El anterior proceso se observa en la *Figura 4.27*.

```
Iniciar Proceso  
IdBus[2]: Char  
IdBus[2] ← {"0"}  
flat, flon, falt : float  
gps_lon[20]: char  
gps_lat[20]: char  
Iniciar Loop  
Obtener Latitud, Longitud  
flat ← Latitud  
flon ← Longitud  
gps_lon ← flon  
gps_lat ← flat  
Concatenar gps_lon ← gps_lon, gps_lat, IdBus  
Enviar gps_lon  
Fin Loop  
Fin Proceso
```

*Figura 4. 27 Pseudocódigo para la obtención de ubicación del bus en coordenadas GPS*

#### 4.3.2.2. Implementación software Módulo Gateway

El MG recibe paquetes comprimidos en una variable llamada *bufer*, la cual se descomprime con el fin de obtener datos de latitud y longitud requeridos por el MS; se establece la comunicación IP entre el MG y el MS y se hace transferencia de datos aplicando el comando Linux *curl*, manejado por la Gateway OLG01, en conjunto de solicitudes API RESTful, que se ejecutan cada vez que se recibe un paquete de datos. El comando *Curl*, es un comando Linux para hacer transferencia de archivos compatible con varios protocolos de comunicación y

hace uso de solicitudes API RESTful para determinar el tipo de transferencia de datos. Al mismo tiempo que se desarrolla la transmisión de datos hacia el MS se deben concatenar los datos de longitud, latitud y número de ruta descomprimidos, con el propósito de enviarlos hacia el MP. Lo descrito anteriormente se muestra en el pseudocódigo de la *Figura 4.28*.

```
Iniciar Proceso  
datasend[50] : uint8_t  
IdBus[20]: char  
lat[20] : char  
alt[20] : char  
buf[50] :uint8_t  
Iniciar Loop  
Descomprimir buf  
Obtener Latitud, Longitud, identificador de bus  
lat ← Latitud  
lon ← Longitud  
IdBus ← identificador de bus  
Iniciar Proceso ejecutar comando LINUX  
Iniciar Proceso Curl  
Enviar lat, lon al servidor  
Fin Proceso Curl  
Fin Proceso ejecutar comando LINUX  
Fin Loop  
Fin Proceso
```

*Figura 4. 28 Pseudocódigo proceso en el MG*

#### 4.3.2.3. Implementación software Módulo Parada

El MP descomprime los elementos de latitud, longitud y número de ruta contenidos en la variable *buffer* para realizar los cálculos definidos en el diseño, con el fin de proporcionar información de tiempo de espera en la pantalla LCD. La librería

LiquidCrystal permite el control sobre la funcionalidad de la pantalla para el despliegue de información en parada. La *Figura 4.29* expone el pseudocódigo realizado en el MP.

```
Iniciar Loop  
Descomprimir buf  
Obtener Latitud, Longitud, número de ruta  
lat←Latitud  
lon←Longitud  
IdBus← Identificador de bus  
IdB← IdBus  
  Sí (IdB=1)  
    Convertir dato coordenada Latitud y Longitud a formato en metros  
    Encontrar distancia del bus a la parada  
    d← distancia del bus a la parada  
    Sí (d>=20 y Latitud bus<Latitud parada)  
      Encontrar Tiempo de espera del bus a la parada  
      tiempo ← Tiempo de espera del bus a la parada  
      Escribir LCD (IdBus)  
      Escribir LCD (tiempo en minutos:segundos)  
    Fin Sí  
    Si – No  
      Escribir LCD (El bus ha llegado)  
    Fin Sí  
    Si – No  
      Escribir LCD (El bus no corresponde)  
Fin Loop  
Fin Proceso
```

*Figura 4. 29 Pseudocódigo del proceso realizado en MP*

#### 4.3.2.4. Implementación software Módulo Servidor

En el servidor ThingSpeak se crea un canal con el nombre de *Localización buses*, sobre el cual se desarrollan dos campos correspondientes a *Latitud* y *Longitud*, encargados de recibir los datos enviados desde el MG por medio del comando

Linux *curl*. Con el fin de actualizar los campos del canal *Localización buses* se emplea el comando *curl* junto a la solicitud API; el protocolo de comunicación utilizado en esta transferencia de archivos corresponde al HTTP.

Se hace uso de ID del canal y una READ API KEY para gestionar los datos del servidor en una página Web y una WRITE API KEY para la actualización de los campos *Latitud* y *Longitud* en el canal *Localización buses*.

En el servidor Google Maps se crea un proyecto con el nombre de *Sistema de Localización* y se obtiene una API KEY para gestionar los mapas del servidor sobre una página Web. Posteriormente se desarrolla una página Web con el uso del lenguaje de programación HTML y Java Script, con el propósito de obtener los datos de latitud y longitud del servidor ThingSpeak; de igual forma se extraen los mapas desde el servidor de Google Maps por medio de la API KEY obtenida en Google Cloud Platform. Una vez desarrollada la página Web, el usuario puede visualizar en tiempo real la ubicación del bus. Para el desarrollo de la página Web se plantea el pseudocódigo mostrado en la *Figura 4.30*.

```
Inicio  
Cargar API de JavaScript de Google Maps  
Obtener Datos JSON correspondientes a latitud  
X ← latitud  
Obtener Datos JSON correspondientes a longitud  
Y ← longitud  
Cargar Mapa  
Fijar Punto(X,Y)  
Fin
```

*Figura 4. 30 Pseudocódigo para el desarrollo de la página Web*

#### 4.3.3. Implementación hardware

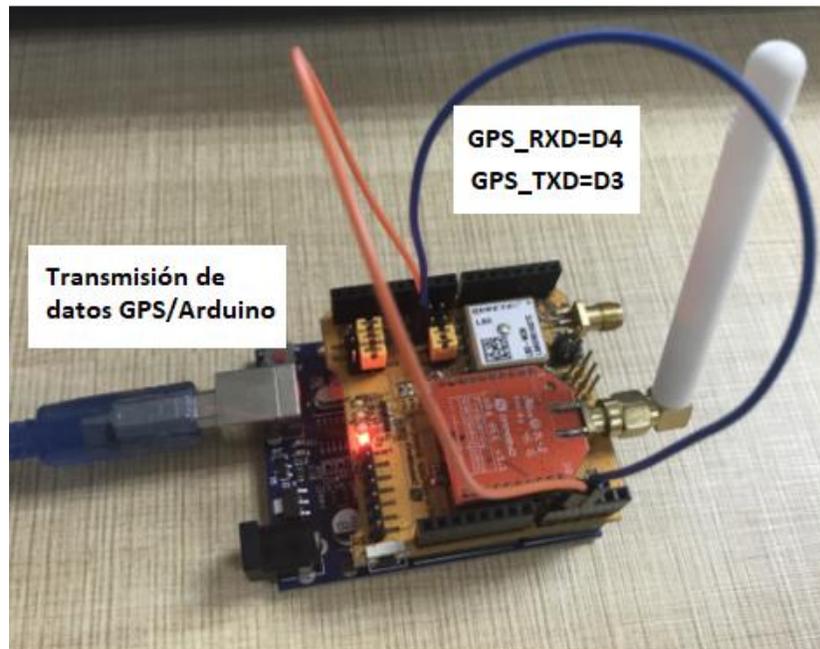
A continuación se detalla la implementación hardware, donde se llevan a cabo los procesos de conexión entre los diferentes dispositivos.

#### 4.3.3.1. Implementación hardware Módulo a Bordo

En este módulo se realiza la interconexión del dispositivo LoRa/GPS con la tarjeta Arduino, de donde se remueve el puente que existe entre los pines GPS\_RXD y GP\_TXD, habilitando así, la transmisión y recepción de datos entre los dispositivos como se observa en las Figuras 4.31 y 4.32.

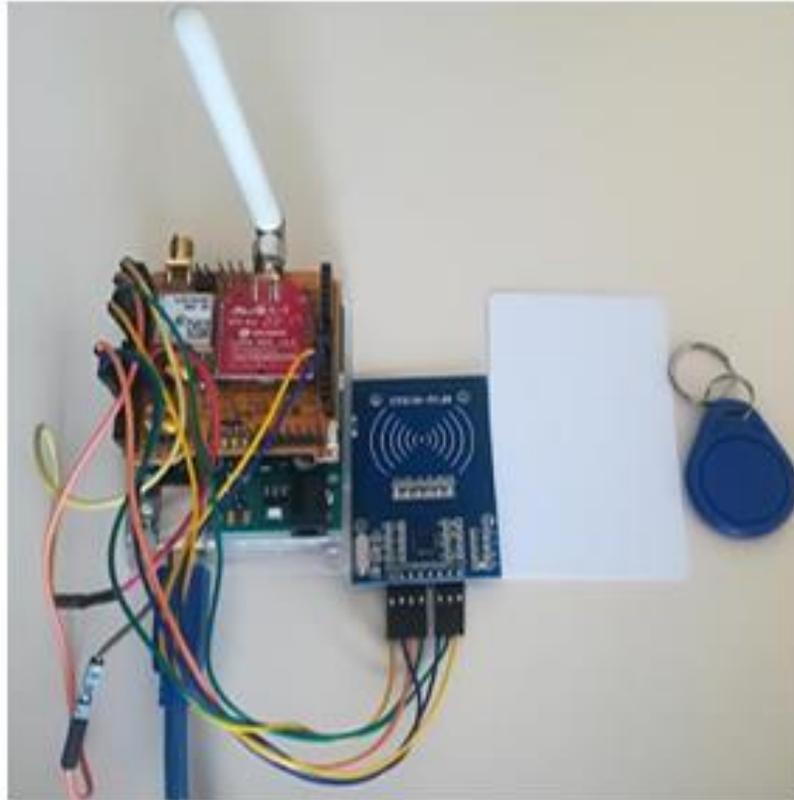


**Figura 4.31** Conexión LoRa/GPS Shield V95 con el Arduino uno.



**Figura 4.32** Conexión de transmisión y recepción de datos entre GPS y Arduino uno.

Se procede a conectar el módulo RFID y el Arduino en base a la asignación de pines de la *Tabla 4.3*. En la *Figura 4.33* se observa la implementación hardware del MaB en su totalidad.



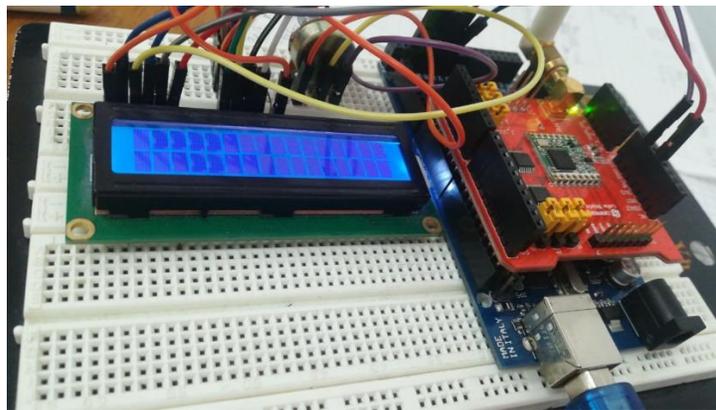
***Figura 4.33 Implementación hardware MaB.***

#### 4.3.3.2. Implementación hardware Módulo Parada

Los dispositivos LoRa Shield V95 y Arduino uno son ensamblados para permitir transmisión y recepción de datos en conjunto con la implementación software descrita anteriormente, como se puede observar en la *Figura 4.34*. Se procede a conectar la LCD con el Arduino, siguiendo la asignación de pines descrita en la *Tabla 4.4*, con el fin de permitir procesos de control sobre la pantalla. En la *Figura 4.35* se percibe la implementación final del MP.



**Figura 4. 34 Dispositivos LoRa Shield V95 y Arduino UNO acoplados**



**Figura 4.35 Implementación hardware MP culminada.**

Finalmente el piloto de información desarrollado, permite determinar el tiempo de espera en las 3 paradas establecidas; cada parada cuenta con una pantalla LCD, en la cual se muestra esta información y el número de ruta. La comunicación entre módulos para realizar este proceso, se realizó mediante la tecnología LoRa, disponiendo de la documentación necesaria acerca de la configuración de cada equipo, lo que facilitó los procesos de desarrollo para dar funcionalidad a cada

módulo. En la *Figura 4.36* se observa la presentación al usuario mediante la pantalla LCD en un punto del escenario, donde se visualiza la ruta 1 y un tiempo de espera igual a 1 minuto y 15 segundos.



**Figura 4.36 Presentación de la información al usuario mediante LCD**

Conjuntamente el piloto despliega la ubicación del bus en tiempo real mediante una página Web, gracias a los servidores de ThingSpeak y Google Maps, facilitando la visualización a través de mapas y de imágenes satelitales. Las *Figuras 4.37* y *4.38* presentan la ubicación del bus en un instante de tiempo, mediante el punto de localización rojo, esta información es valiosa para los usuarios, ya que permite observar gráficamente el lugar donde se encuentra el bus en todo momento.



**Figura 4.37 Visualización en mapa de la página Web**



**Figura 4. 38** Visualización satelital en la página Web

#### 4.4. PRUEBAS Y ANÁLISIS DE RESULTADOS

Se definen las pruebas de desempeño del piloto y su ejecución. La evaluación se realiza mediante pruebas de confiabilidad y cobertura.

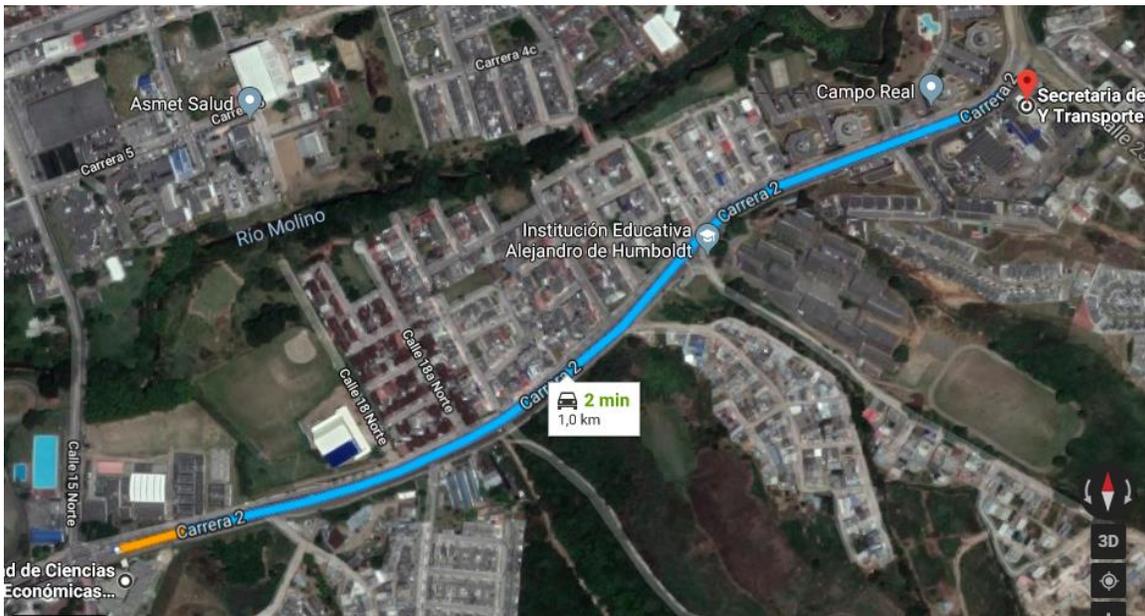
**Confiabilidad:** La confiabilidad de un sistema de comunicación se refiere a que este cumpla con el propósito para el cual fue creado, bajo condiciones de uso definidas. La prueba de confiabilidad permite determinar si el tiempo de espera que proporciona el piloto en cada parada, es próximo a un tiempo teórico y a un tiempo medido por cronómetro [74].

**Cobertura:** La cobertura en un sistema de comunicación corresponde al área geográfica dentro de la cual, el usuario o dispositivo puede establecer comunicación con otros nodos o con el centro de control, logrando el acceso a la red. El propósito de esta prueba es determinar el radio máximo de cobertura del MG sobre el escenario establecido, definiendo la intensidad de señal en cada espacio.

##### 4.4.1. Elección del escenario

El escenario se establece sobre la carrera segunda, Avenida Pomona de la ciudad de Popayán, abarcando un tramo aproximado a un kilómetro, en la *Figura 4.39* se observa el escenario, mediante la línea azul, iniciando el trayecto desde la Facultad de Ingenierías de la Universidad del Cauca hasta la Secretaria de

Tránsito y Transporte de la ciudad de Popayán. La Avenida Pomona cuenta con dos carriles, uno de sur a norte y otro de norte a sur.



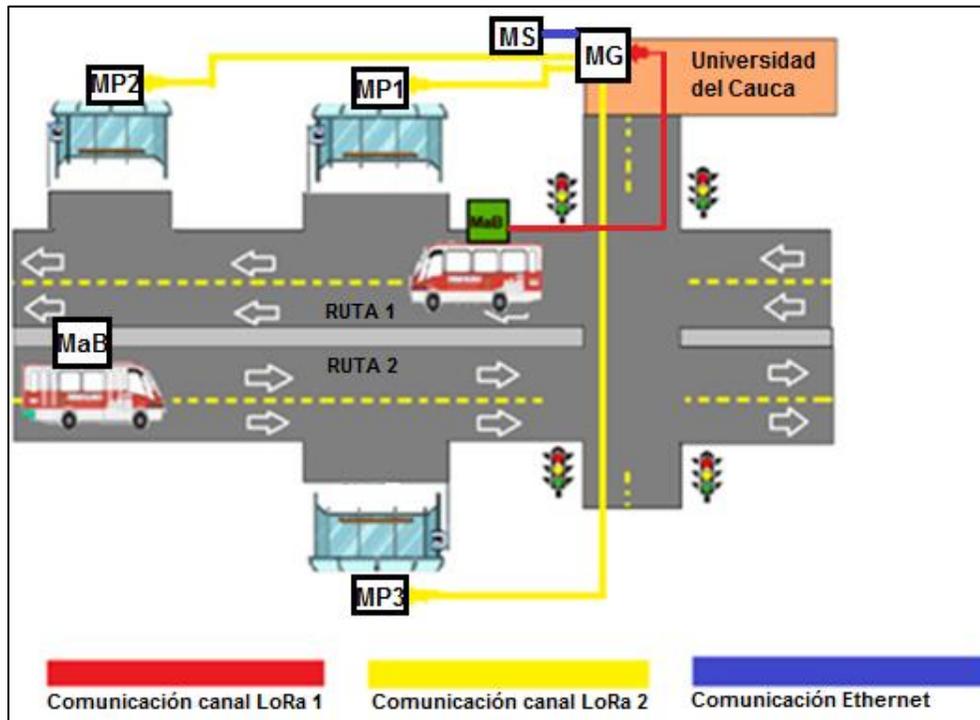
**Figura 4.39 Avenida Pomona (escenario de pruebas).**

En el escenario se definen tres paradas, la primera, denominada P1, ubicada aproximadamente a 300 metros de la Facultad de Ingenierías sobre el carril de occidente a oriente; la segunda, denominada P2, se ubica sobre el mismo carril aproximadamente a 400 metros de la primera, alcanzando una distancia de 700 metros desde la Facultad de Ingenierías; la tercera, denominada P3, situada diagonal a la primer parada, sobre el carril opuesto (oriente a occidente), alcanzando 850 metros medidos desde la Secretaria de Tránsito y Transporte de Popayán. En la *Figura 4.40* se observa la ubicación de P1, P2 y P3 mediante los puntos de localización amarillo, verde y rojo respectivamente.



**Figura 4.40. Ubicación de las paradas.**

En la *Figura 4.41* se observa el diagrama esquemático del escenario y la comunicación entre módulos a través de la tecnología LoRa y Ethernet.



**Figura 4.41 Diagrama esquemático del escenario.**

Para la realización de pruebas, es necesario el uso de dos rutas diferentes; la ruta 1 recorre el trayecto desde la Facultad de Ingenierías hasta la Secretaria de Tránsito y Transporte de la ciudad de Popayán (occidente a oriente) y la ruta 2 recorre el trayecto de oriente a occidente, con el fin de realizar un proceso de identificación de ruta en cada parada, si la ruta le corresponde a la parada, se efectúa el análisis de los datos, de lo contrario, se descartan.

La herramienta Google Maps es necesaria para la visualización satelital del escenario, también es conveniente para la obtención de las coordenadas<sup>25</sup> de cada parada, las cuales se observan en la *Tabla 4.5*.

	Latitud	Longitud
<b>Parada 1</b>	2.447717	-76.595985
<b>Parada 2</b>	2.449729	-76.592734
<b>Parada 3</b>	2.447800	-76.596321

*Tabla 4. 5 Coordenadas geográficas de las paradas.*

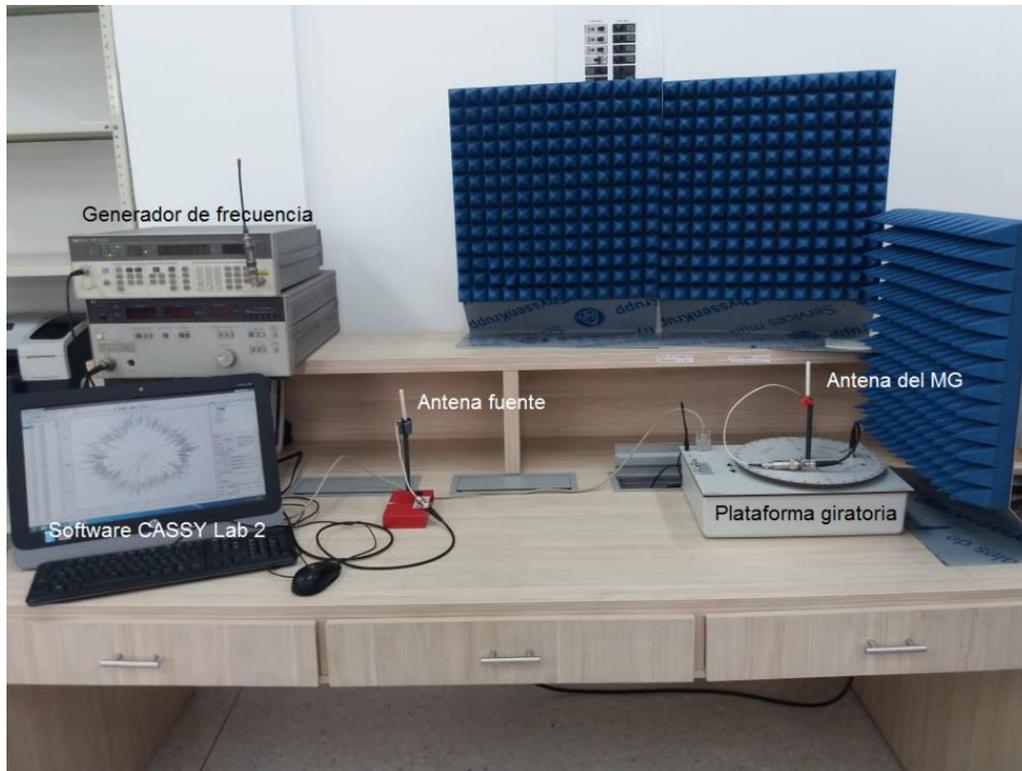
#### 4.4.2. Configuración del escenario

Para establecer la ubicación de los módulos, en especial del MG, se debe contar con información apropiada sobre las características de la antena del dispositivo implementado, con el fin de situarlo de tal forma que se logre un buen desempeño del piloto; por falta de información de dichas características, se realiza un análisis de simulación para determinar el tipo de antena y su patrón de radiación.

Se emplea una antena fuente configurada en 915 Mhz mediante un generador de frecuencia y la antena de prueba (antena del MG) ubicada sobre una plataforma giratoria, la cual permite determinar el patrón de radiación, que se visualiza a través del software CASSY Lab 2. *En la Figura 4.42* se observa el montaje descrito anteriormente.

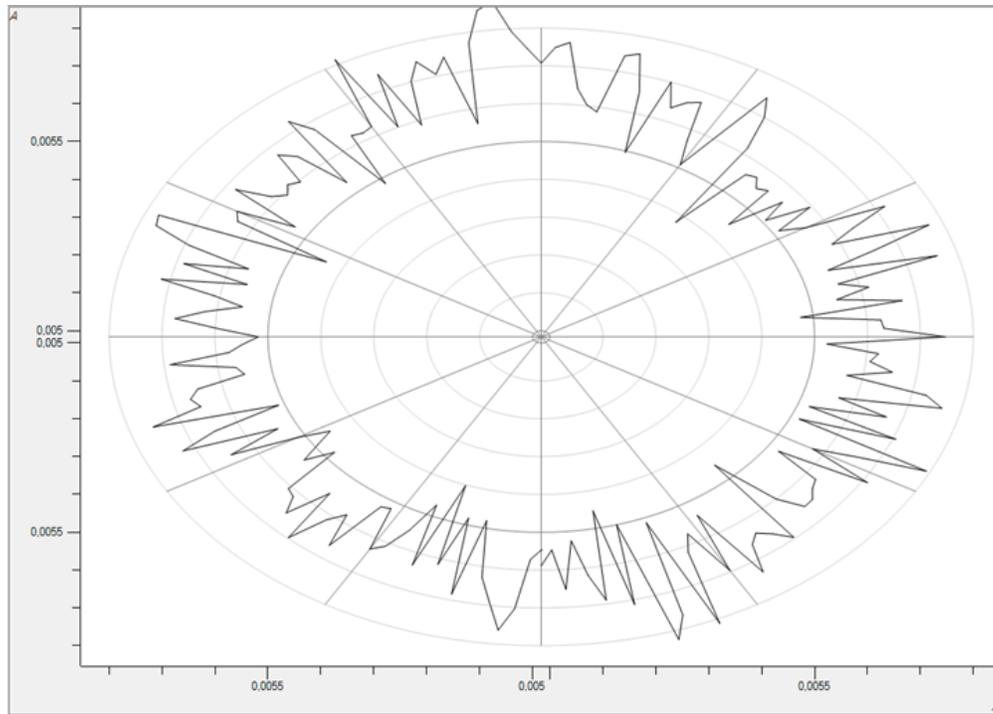
---

<sup>25</sup> Las coordenadas arrojadas por la herramienta Google Maps, corresponden a coordenadas geográficas decimales, el mismo tipo de coordenadas que arroja el GPS implementado

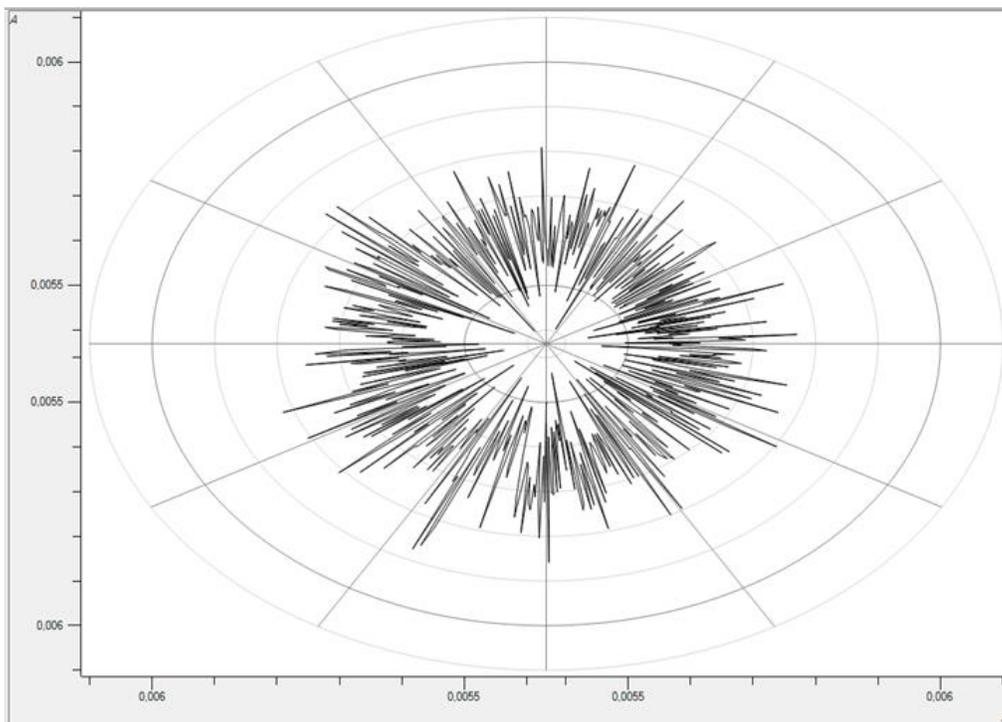


**Figura 4.42 Montaje para obtener el patrón de radiación de la antena del MG.**

En las Figuras 4.43 y 4.44 se observa el resultado correspondiente al patrón de radiación de la antena del MG.



**Figura 4.43 Patrón de radiación de la antena del MG con paso angular  $\approx 2^{\circ 26}$ .**



**Figura 4.44 Patrón de radiación de la antena del MG con paso angular  $= 0.5^{\circ}$ .**

<sup>26</sup> El paso angular determina los intervalos de giro de la plataforma giratoria, los cuales pueden ser:  $0.5^{\circ}$ ,  $1^{\circ}$  o  $2^{\circ}$ .

Las *Figuras 4.43 y 4.44* representan una aproximación del patrón de radiación de una antena omnidireccional, se consideran algunas alteraciones en la medición, debido a las interferencias generadas por el entorno donde fue desarrollado y a las reflexiones en la señal dadas por el medidor, ya que no se contaba con una cámara anecoica<sup>27</sup>, sin embargo, se colocaron paneles alrededor de la antena de prueba para aminorar dichas alteraciones.

Los dispositivos LoRa utilizados en MaB, MG y MP, cuentan con el mismo tipo de antena helicoidal de diámetro próximo a los 4mm, por lo que se define que operan en el modo normal ya que la longitud de onda es mucho mayor que las dimensiones de las antenas. Para este modo de operación las antenas describen un comportamiento omnidireccional [75].

Se procede a ubicar cada módulo sobre el escenario de la siguiente manera:

- MaB: Se ubica dentro de un automóvil, el cual representa el bus y se moviliza a una velocidad constante (30Km/h), sobre las rutas definidas.
- MG: Se ubica sobre la azotea de la Facultad de Ingenierías de la Universidad del Cauca, ya que requiere de un lugar alto, desde el cual pueda tener visibilidad con el MaB y los MP (La Gateway utiliza una antena omnidireccional, por lo cual no es indispensable fijar una orientación para el dispositivo). El edificio cuenta con más de 15 metros de altura, siendo el más alto en este sector y es de fácil acceso por parte del cuerpo estudiantil, lo cual es apropiado para realizar las pruebas del piloto.
- MP: Se ubican sobre las coordenadas geográficas expuestas en la *Tabla 4.5*. En el escenario real, estos puntos cuentan con una bahía, facilitando el estacionamiento del automóvil donde se moviliza el MaB, haciéndolos apropiados para las paradas del piloto.
- MS: Se ubica cerca al MG, ya que se comunican a través de Ethernet.

#### 4.4.3. Prueba de confiabilidad del piloto

La prueba de confiabilidad consiste en medir el tiempo de espera en cada parada mediante el piloto y un cronómetro, para realizar una comparación entre estos resultados y un valor teórico, con el propósito de determinar la precisión del tiempo de espera en parada.

---

<sup>27</sup> Es una cámara aislada del exterior, evitando que entre en ella algún tipo de ruido u onda electromagnética que pueda causar interferencia.

Inicialmente se determina el valor teórico ( $t_t$ ) de tiempo de espera, considerando un movimiento uniforme, para ello se emplea la siguiente ecuación:

$$t = d/v \quad (4.1)$$

Donde  $v = 30Km/h$ , los resultados se observan en la *Tabla 4.6*.

	Tiempo de espera teórico ( $t_t$ )
<b>Parada 1</b> (300 metros)	36 segundos (0:36)
<b>Parada 2</b> (700 metros)	1 minuto y 24 segundos (1:24)
<b>Parada 3</b> (850 metros)	1 minuto y 42 segundos (1:42)

*Tabla 4.6 Valor de tiempo de espera teórico de cada parada*

Para determinar los tiempos de espera del piloto y los medidos por cronómetro, es necesario que el MaB recorra las rutas establecidas a una velocidad constante  $v = 30Km/h$ , con el fin de calcular el tiempo de llegada del MaB a cada MP (el tiempo de espera de P1 y P2, se calcula desde la Facultad de Ingenierías de la Universidad del Cauca y el tiempo de espera de P3, se calcula desde la Secretaria de Tránsito y Transporte de Popayán); este proceso se realiza 50 veces.

Considerando que el vehículo en el que se desplaza el MaB, no tiene una velocidad instantánea, es apropiado estimar un tiempo de retardo, dentro del cual, el vehículo alcanza los  $30Km/h$ ; de las 50 repeticiones se establece un valor promedio de retardo igual a 7 segundos, al cual se le denomina tiempo de retardo por arranque ( $t_a$ ).

Además se realiza un análisis del GPS LoRa utilizado ya que al tener un margen de error, puede afectar el cálculo del tiempo de espera en cada MP; por lo anterior se realiza una prueba que permita determinar una aproximación de este error y el retardo que genera. Para ello se emplea el dispositivo *Magellan MobileMapper 6 GPS*, el cual recopila datos de posicionamiento con una precisión en tiempo real de 1 a 3 m; se realiza una medición simultánea de las coordenadas en diferentes puntos mediante los dos GPS, el dispositivo Magellan se considera como referencia, esto se observa en la *Figura 4.45* [76].



**Figura 4.45 Prueba del margen de error del GPS LoRa.**

Al obtener las coordenadas proporcionadas por los dispositivos GPS, en un mismo punto, se realiza el cálculo de distancia entre dos puntos mediante el Teorema de Pitágoras<sup>28</sup>. Las coordenadas obtenidas por cada dispositivo y el cálculo de distancia entre los puntos se observa en la *Tabla 4.7*.

<sup>28</sup> La distancia (d) entre dos puntos P1 (x1,y1) y P2 (x2,y2) se determina mediante la expresión  $d = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$ .

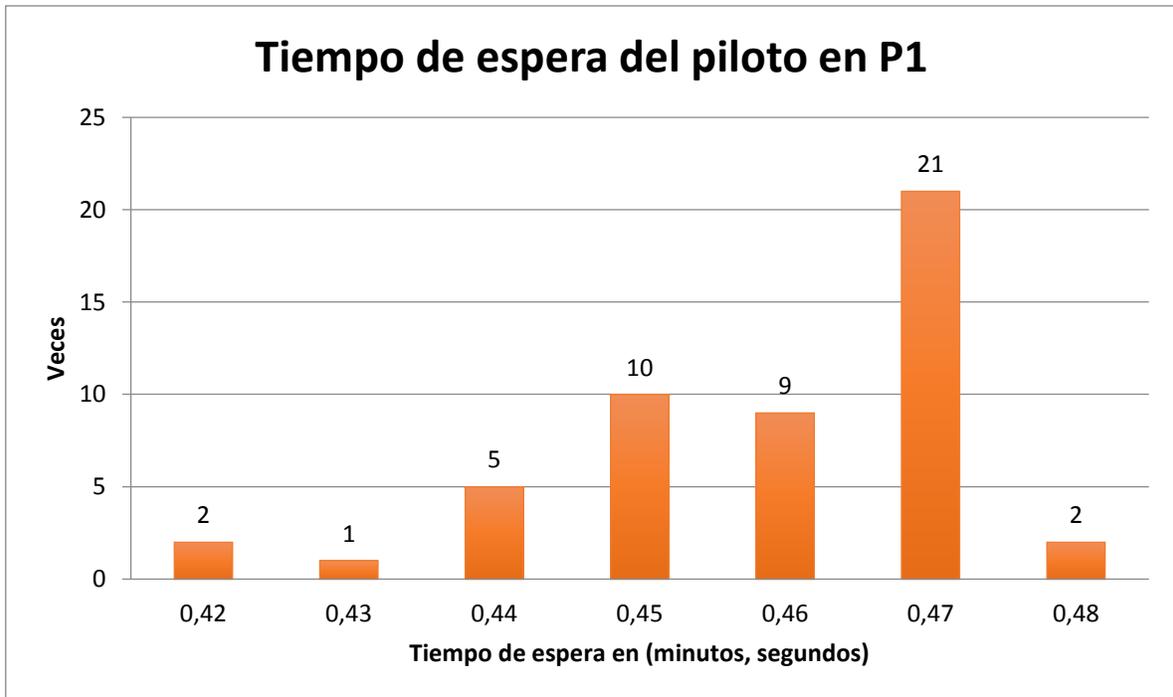
Coordenadas GPS Magellan (Grados decimales)	Coordenadas GPS LoRa (Grados decimales)	Distancia de separación (m)
Latitud: 2.446398 Longitud: -76.597964	Latitud: 2.4464444 Longitud: -76.597972	5.211548889
Latitud: 2.4464906 Longitud: -76.5986131	Latitud: 2.446525 Longitud: -76.598595	4.303621428
Latitud: 2.446574 Longitud: -76.597986	Latitud: 2.446538 Longitud: -76.597984	3.987115339
Latitud: 2.447511 Longitud: -76.597595	Latitud: 2.447495 Longitud: -76.597633	4.581639911
Latitud: 2.446675 Longitud: -76.598013	Latitud: 2.446718 Longitud: -76.598029	5.076818357
Latitud: 2.446153 Longitud: -76.598487	Latitud: 2.446195 Longitud: -76.598507	5.149369253
Latitud: 2.447305 Longitud: -76.598314	Latitud: 2.447272 Longitud: -76.598352	5.583587582
Latitud: 2.445825 Longitud: -76.598621	Latitud: 2.445854 Longitud: -76.598576	5.94411960933
Latitud: 2.445735 Longitud: -76.598428	Latitud: 2.445749 Longitud: -76.598396	3.88128177409
Latitud: 2.446904 Longitud: -76.598801	Latitud: 2.446861 Longitud: -76.598819	5.15908567963

**Tabla 4.7 Comparación de coordenadas del GPS de referencia y el GPS LoRa**

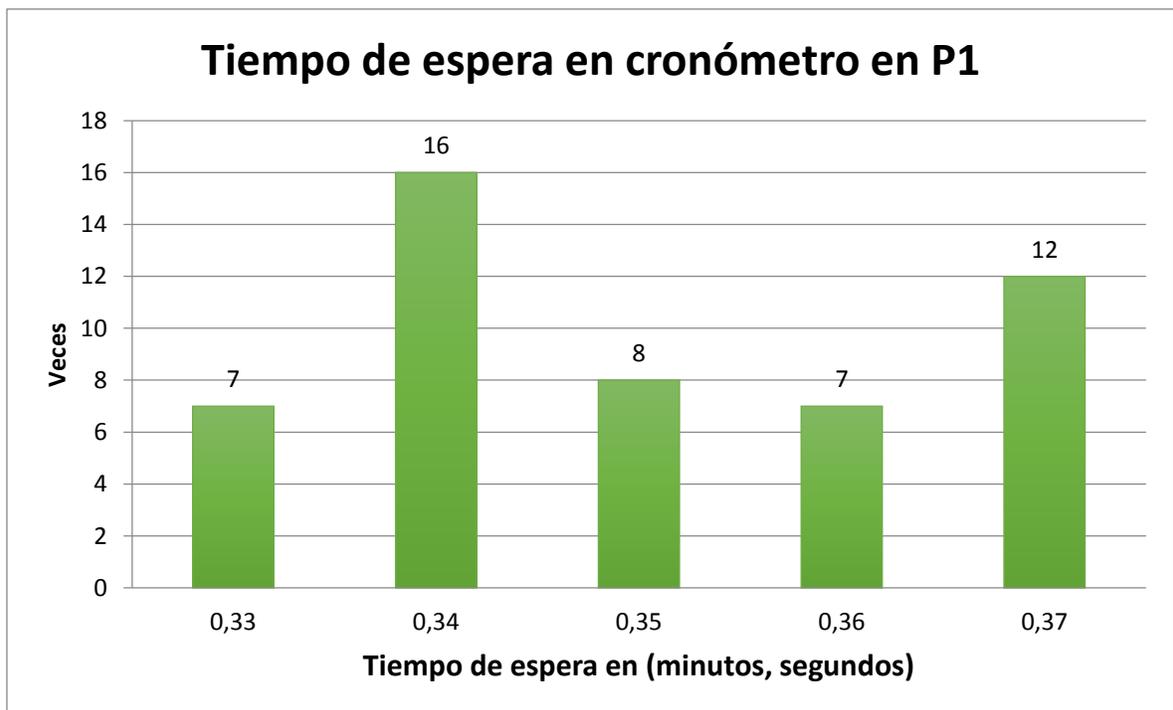
De los datos de distancia entre los dos GPS consignados en la *Tabla 4.7*, se obtiene una Media igual a 4,85184877 m, este valor genera un tiempo de retardo de GPS ( $t_{gps}$ ) aproximado a 0,582222 segundos.

Los datos  $t_a$  y  $t_{gps}$  no se tienen en cuenta en el cálculo de tiempo de espera realizado por el piloto, por consecuencia los datos generados presentan un margen de error a considerar.

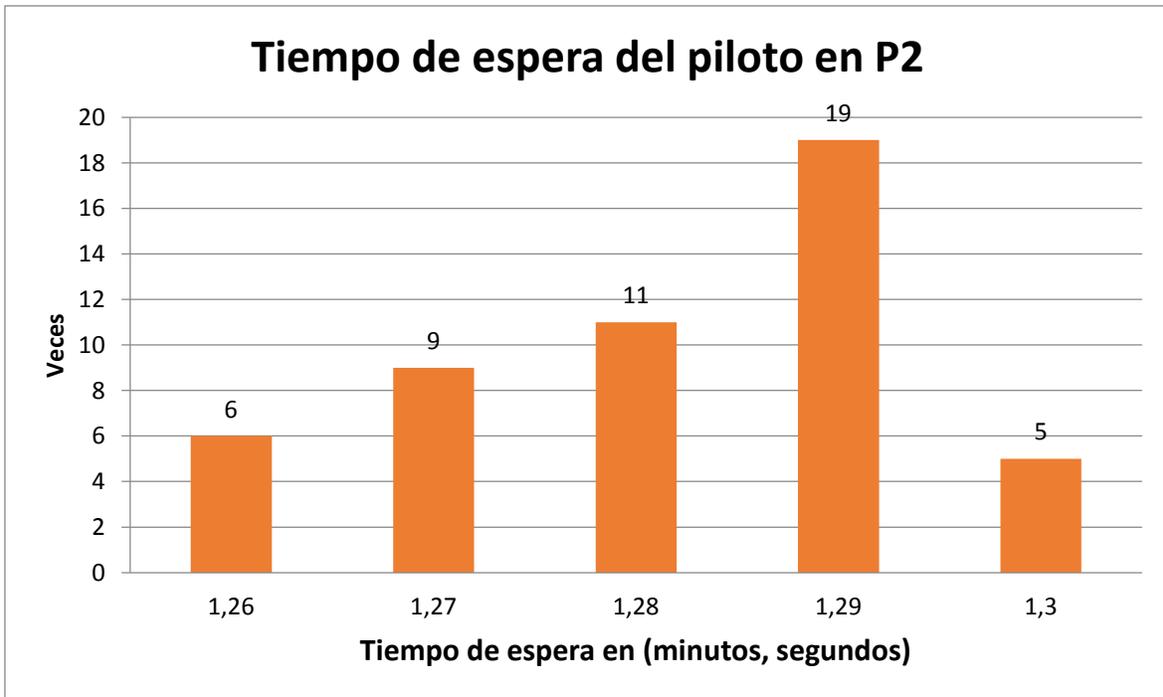
En las siguientes gráficas se observan los resultados de los tiempos de espera del piloto y los medidos por cronómetro (los resultados se expresan en formato de minutos y segundos, donde, tener un valor=1,59, representa 1 minuto y 59 segundos).



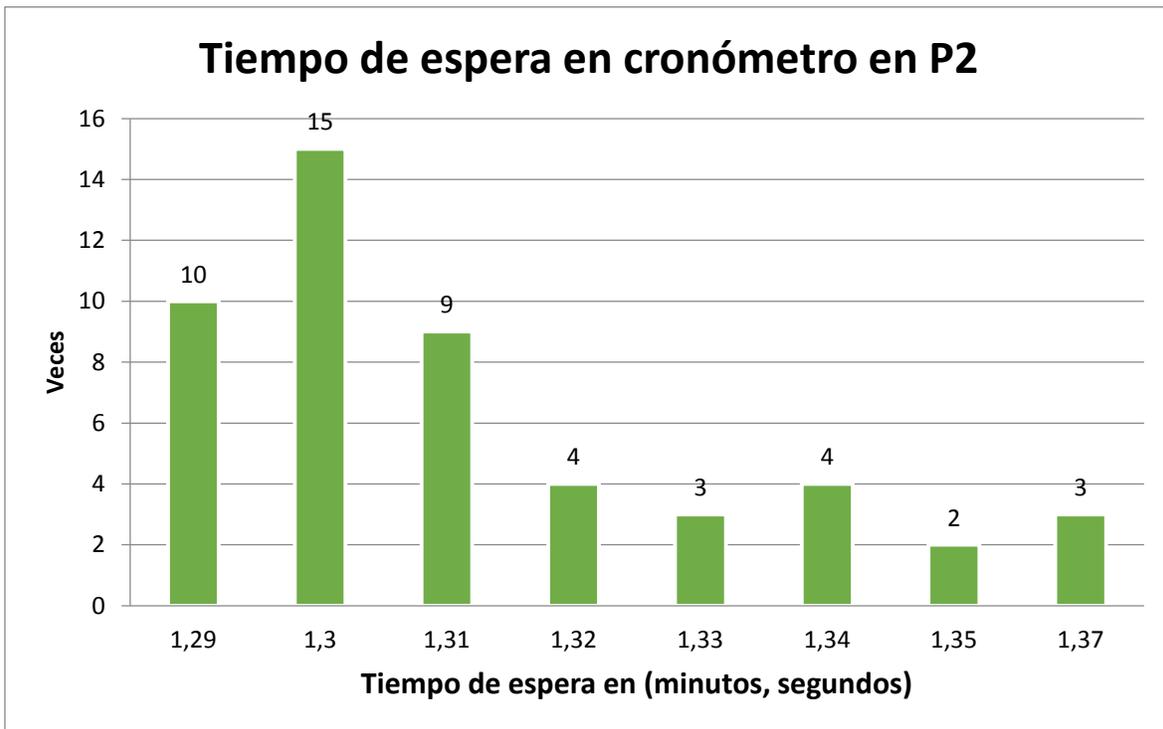
*Figura 4.46 Tiempo de espera del piloto en P1.*



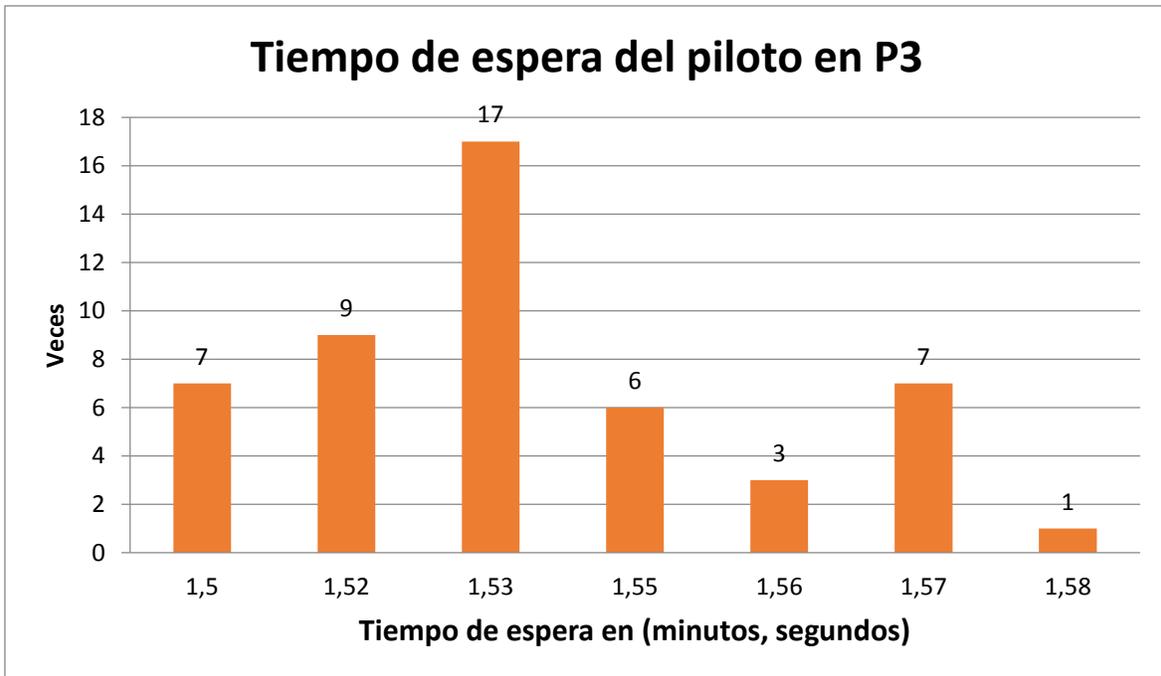
*Figura 4.47 Tiempo de espera medido por cronometro en P1.*



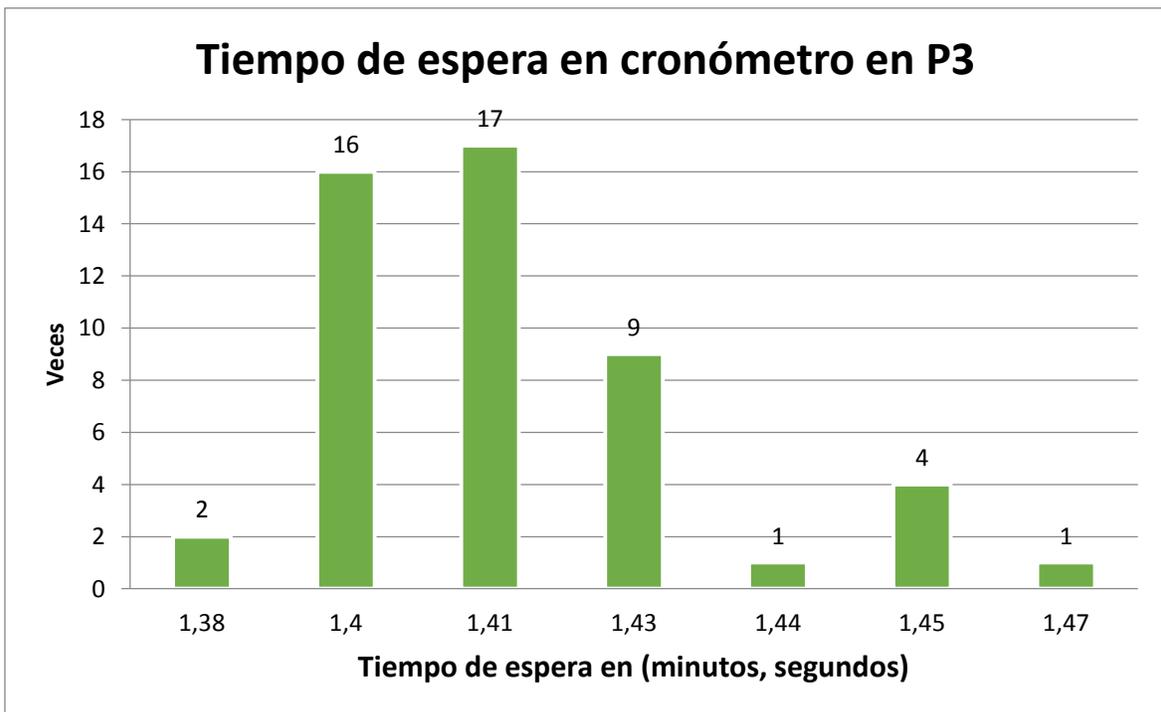
*Figura 4.48 Tiempo de espera del piloto en P2.*



*Figura 4.49 Tiempo de espera medido por cronómetro en P2.*



*Figura 4.50 Tiempo de espera del piloto en P3.*



*Figura 4.51 Tiempo de espera medido por cronómetro en P3.*

En la *Tabla 4.8* se observan los parámetros estadísticos de cada parada (Donde,  $T_p$ =Tiempo de espera del piloto y  $T_c$ =Tiempo de espera por cronómetro).

	<b>P1 Tp (min,seg)</b>	<b>P1 Tc (min,seg)</b>	<b>P2 Tp (min,seg)</b>	<b>P2 Tc (min,seg)</b>	<b>P3 Tp (min,seg)</b>	<b>P3 Tc (min,seg)</b>
<b>Media</b>	0,4588	0,3502	1,2816	1,3126	1,5348	1,4142
<b>Mediana</b>	0,46	0,35	1,28	1,305	1,53	1,41
<b>Moda</b>	0,47	0,34	1,29	1,3	1,53	1,41
<b>Desviación estándar</b>	0,01423419	0,01421267	0,0120136	0,02229762	0,02251893	0,01863615

**Tabla 4.8 Parámetros estadísticos del tiempo de espera en cada parada.**

A la Media del piloto ( $t_{mp}$ ), registrada en la *Tabla 4.8*, se le restan los retardos  $t_{gps}$  y  $t_a$ , con la finalidad de encontrar un valor promedio del tiempo de espera del piloto ( $t_{vp}$ ) bajo las mismas características de los tiempos medidos por cronómetro, obteniendo la ecuación:

$$t_{vp} = t_{mp} - t_{gps} - t_a \quad (4.2)$$

Además se encuentra la velocidad promedio ( $V_p$ ), a la que se desplazaba el MaB, usando la Media dada por el cronómetro ( $t_c$ ); este cálculo se realiza mediante la ecuación 4.1.

- Parada 1:

Mediante la ecuación 4.2 se determina  $t_{vp1}$ :

$$t_{vp1} = 46 \text{ s} - 7 \text{ s} - 0.58 \text{ s} = 38.42 \text{ segundos} \quad (4.3)$$

$$t_{vp1} = 0,38 \text{ (formato minutos, segundos)}$$

Despejando  $v$  de la ecuación 4.1, se calcula  $V_{p1}$ :

$$V_{p1} = \frac{300m}{35s} = \frac{8.57m}{s} = \frac{30.852Km}{h} \quad (4.4)$$

- Parada 2:

El  $t_{vp2}$  es:

$$t_{vp2} = 88 \text{ s} - 7 \text{ s} - 0.58 \text{ s} = 80.42 \text{ segundos} \quad (4.5)$$

$$t_{vp2} = 1,20 \text{ (formato minutos, segundos)}$$

$V_{p2}$  es igual a:

$$V_{p2} = \frac{700m}{91s} = \frac{7.69m}{s} = \frac{27.69Km}{h} \quad (4.6)$$

- Parada 3:

El  $t_{vp3}$  es igual a:

$$t_{vp3} = 113s - 7s - 0.58s = 105.42 \text{ segundos} \quad (4.7)$$

$$t_{vp3} = 1,45 \text{ (formato minutos, segundos)}$$

$V_{p3}$  es igual a:

$$V_{p3} = \frac{850m}{101s} = \frac{8.41m}{s} = \frac{30.29Km}{h} \quad (4.8)$$

La *Tabla 4.9* presenta el comparativo entre los tiempos de espera  $t_t$ ,  $t_c$  y  $t_{vp}$  en cada parada.

	$t_t$ (min,seg)	$t_c$ (min,seg)	$t_{vp}$ (min,seg)
<b>Parada 1 (P1)</b>	0,36	0,35	0,38
<b>Parada 2 (P2)</b>	1,24	1,31	1,20
<b>Parada 3 (P3)</b>	1,42	1,41	1,45

**Tabla 4.9 Comparativo entre  $t_t$ ,  $t_c$  y  $t_{vp}$**

De la *Tabla 4.9* se aprecia que P1 y P3 tienen un comportamiento similar, donde  $t_t$  y  $t_c$  varían un segundo y  $t_{vp}$  es mayor 3 y 4 segundos respectivamente con respecto a  $t_c$ , por lo que se deduce que para P1 y P3, el piloto determina el tiempo de espera con un margen de error de 4 segundos. En las dos paradas el valor promedio de velocidad superó levemente los  $30Km/h$ , por consecuencia  $t_c$  es menor a  $t_t$  y  $t_{vp}$ ; esta variación en la velocidad es causa del tráfico existente durante la realización de la prueba. Para P2 la velocidad promedio es menor a los  $30Km/h$  establecidos, esto afectó el  $t_c$ , obteniendo un tiempo de espera mayor en comparación con  $t_t$  y  $t_{vp}$ ; para esta parada  $t_{vp}$  está más cerca al tiempo de espera teórico a 4 segundos, lo que coincide con la variación en el cálculo del

tiempo de espera presentado en P1 y P3, por lo cual se concluye que el piloto tiene un margen de error próximo a los 4 segundos.

En la *Tabla 4.8* se observa que la moda de P2 en  $t_p$  y  $t_c$  varía tan solo un segundo y que la desviación estándar es menor, en contraste con P1 y P3, por lo que se deduce que el piloto mide con mayor exactitud el tiempo de espera en esta parada, esto puede ocasionarse por las características del terreno, donde el MP tiene mayor visibilidad con la Gateway a diferencia de P1 y P3 donde posiblemente la señal no llega directamente por línea de vista sino por reflexión, efecto generado por el edificio situado en frente de la azotea donde se ubicó el MG.

En los cálculos realizados para encontrar el valor promedio de tiempo de espera del piloto ( $t_{vp}$ ), se tiene en cuenta que el error del GPS y la velocidad de arranque, altera los 30Km/h definidos para el cálculo de tiempo de espera en el piloto, sin embargo, los resultados permiten concluir que la variación del tiempo de espera calculado y desplegado en cada parada, se proporciona en el orden de los segundos, muy por debajo del minuto, dando un resultado satisfactorio ya que si se asemeja a un Sistema de Transporte Público Inteligente, el usuario espera información sobre el tiempo de llegada del bus a la parada en minutos, discriminando los segundos o dicho de otra manera, es igual si el usuario espera en parada por 10 minutos que por 10 minutos y 4 segundos

#### 4.4.4. Prueba de cobertura del piloto

El propósito de evaluar la cobertura de la Gateway, es determinar hasta qué punto se establece comunicación entre el MG y el MaB; por ello, se realiza el cálculo de cobertura a través de la herramienta de simulación Xirio online y mediante pruebas experimentales.

##### 4.4.4.1. Pruebas de cobertura mediante Xirio

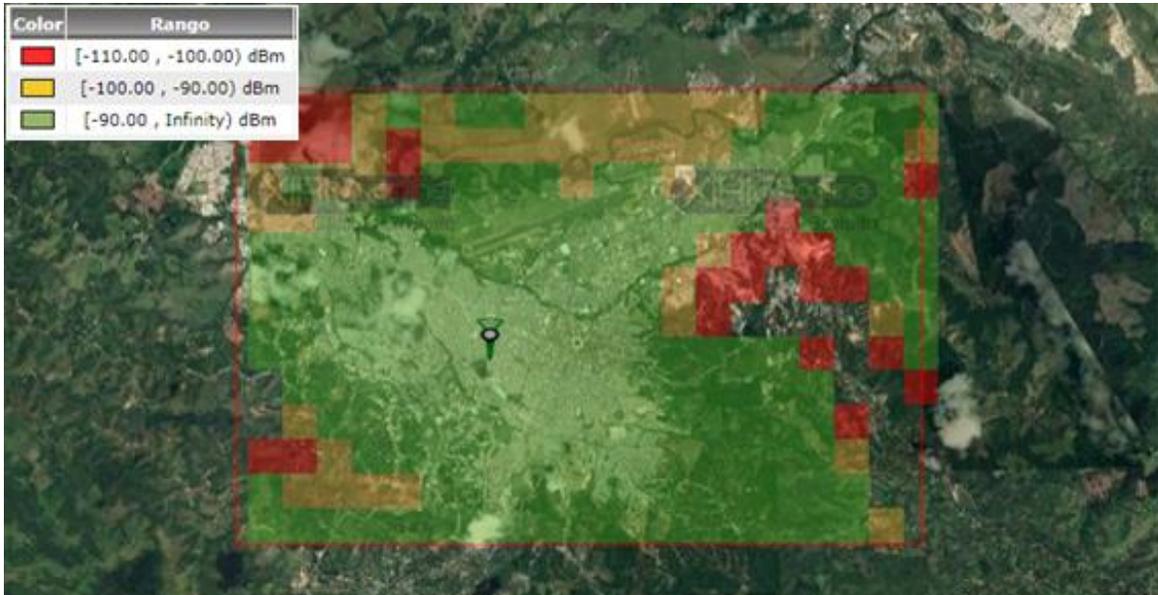
Xirio permite realizar simulaciones profesionales de cobertura radioeléctrica en cualquier parte del planeta; configurando parámetros de transmisión y recepción, de un enlace de radiocomunicación. En la *Tabla 4.10* se observan los parámetros de configuración del estudio de cobertura correspondiente al proyecto en la herramienta Xirio [77].

Parámetros de configuración Xirio
<p><b>Tx (MG):</b></p> <ul style="list-style-type: none"> <li>• Antena: LORA Omnidireccional 2.15 dBi</li> <li>• Altura: 15m</li> <li>• Frecuencia: 915 Mhz</li> <li>• Polarización: Horizontal</li> <li>• Potencia: 20 dBm</li> </ul>
<p><b>Rx (MaB):</b></p> <ul style="list-style-type: none"> <li>• Antena: LORA Omnidireccional 2.15 dBi</li> <li>• Altura: 1.5m</li> <li>• Polarización: Horizontal</li> <li>• Umbral de recepción: -148 dBm</li> </ul>
<p><b>Método de cálculo:</b></p> <ul style="list-style-type: none"> <li>• Método de propagación: Rec. UIT-R P.526<sup>1</sup></li> <li>• Permitividad relativa: 30</li> <li>• Conductividad: 0.15 S/m</li> </ul>

**Tabla 4.10 Parámetros de configuración Xirio [9].**

El método de propagación que se implementa para la realización de esta prueba es el Rec. UIT-R P.526-13, este método se basa en difracción, siendo válido para frecuencias mayores de 30 MHz y permite realizar la simulación con cartografía de media o baja resolución ya que es la única disponible [77].

Los resultados de cobertura del MG mediante la herramienta Xirio se observan en la *Figura 4.52*.



**Figura 4.52 Cobertura del MG con el método Rec. UIT-R P.526-13.**

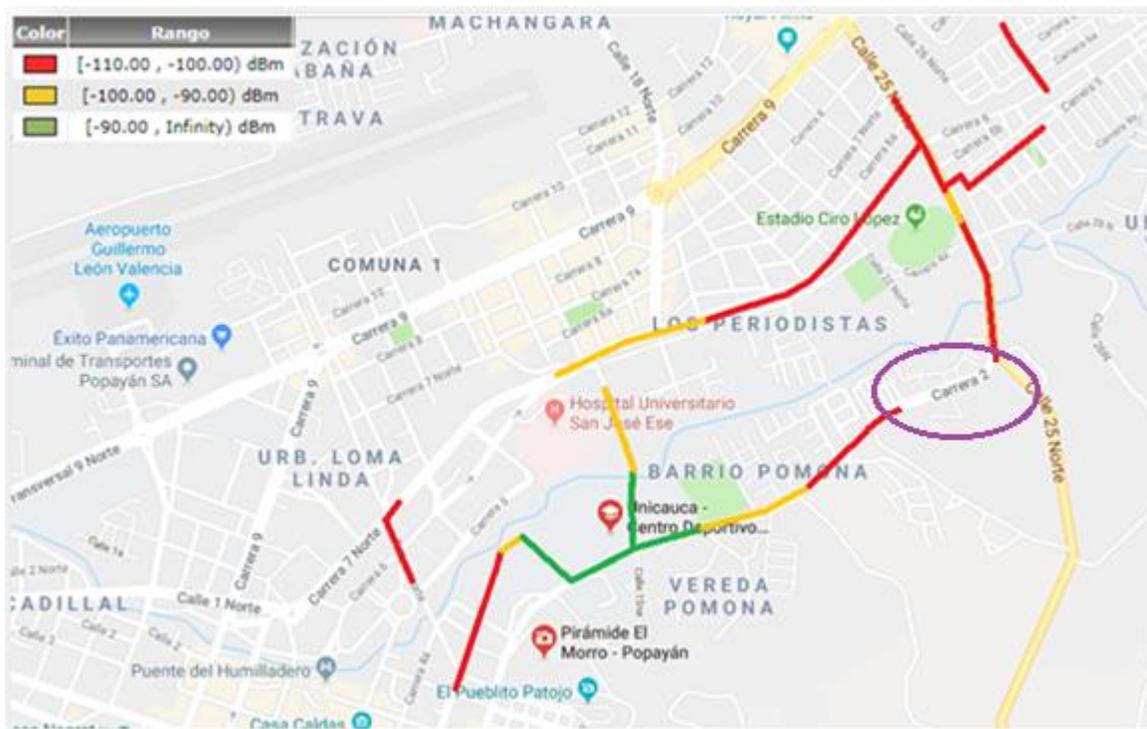
Además, se realiza la simulación con el método Cost 231 como se muestra en la *Figura 4.53*, el cual cuenta con características de terreno más aproximadas al escenario, pero requiere de una cartografía de alta resolución por lo cual no realiza un cálculo de cobertura apropiado.



**Figura 4.53 Cobertura del MG con el método Cost 231.**

#### 4.4.4.2. Pruebas experimentales de cobertura

Se realizan las pruebas experimentales de cobertura, utilizando la configuración 1 descrita en el diseño de comunicación. El MG se encarga de enviar niveles de intensidad de señal al MaB mientras va en movimiento, de esta manera se realiza un mapa de la cobertura máxima del MG, basado en los valores de intensidad de señal recibidos en el MaB. En la *Figura 4.54* se observa, el escenario de prueba y sus alrededores, en el cual se detalla mediante las líneas de color rojo, valores de intensidad de señal entre -110dBm y -100dBm, las líneas color amarillo representan el rango entre -100dBm y -90dBm y las líneas verdes muestran el área donde la intensidad de señal es igual o mayor a -90dBm.



**Figura 4.54** Mapa de cobertura del MG basado en pruebas experimentales.

El cálculo de la cobertura del MG mediante la herramienta Xirio no se muestra con detalle, ya que la cartografía que ofrece no toma en cuenta edificios y obstáculos, los cuales limitan la cobertura del piloto, por lo cual el área que se despliega es mucho más amplia que la cobertura real. Según Xirio los niveles de intensidad de señal hasta -90 dBm abarcan casi todo el territorio de la ciudad de Popayán, lo cual no concuerda con las pruebas experimentales.

Mediante la prueba experimental de cobertura, en la *Figura 4.54* se observan los niveles de intensidad de señal de una forma más detallada, especificando los tramos donde no se obtuvo señal y el área máxima de cobertura del MG, alcanzando un valor aproximado de 2 Km; en coherencia con lo estipulado en el Capítulo 2, donde se especifica que la tecnología LoRa define una cobertura entre 2 y 5 Km para entornos urbanos. Además, permite observar que la cobertura que brinda el MG abarca en gran parte la trayectoria del escenario, visualizando mediante el círculo violeta, que al final de la ruta 1 e inicio de la ruta 2, la señal se pierde.

En general, las pruebas de cobertura generaron diferentes resultados, de los cuales se precisa que la simulación mediante la herramienta Xirio despliega un área de cobertura mucho mayor que el área de cobertura obtenida por las pruebas experimentales, esto se debe a que la cartografía disponible por el software elimina digitalmente los obstáculos, por lo anterior, no se considera una prueba decisiva de cobertura; en contraste con la prueba experimental, la cual define cada espacio por niveles de intensidad de señal, obteniendo un área de cobertura reducida, por lo anterior se evidencia que la cobertura brindada por los dispositivos LoRa se ve limitada por obstáculos como edificios, árboles y características geográficas del terreno, sin embargo, el área de cobertura que proporciona el piloto es buena con respecto al propósito del mismo, ya que la falta de comunicación que se genera entre el MaB y MG en el tramo donde la señal se pierde, no afecta considerablemente el sistema, como se puede observar en el análisis de resultados de confiabilidad, donde el tiempo de espera que despliega el sistema es próximo al tiempo real.

## 5. CAPITULO 5: CONCLUSIONES Y TRABAJOS FUTUROS

En el presente trabajo de grado se expuso el análisis de tecnologías radio para la transmisión de información al usuario de un sistema de control de flota y el diseño e implementación con la tecnología seleccionada. En este capítulo se presentan las conclusiones y los trabajos futuros del trabajo de grado: *Análisis de tecnologías radio para la transmisión de información al usuario de un sistema de control de flota.*

### 5.1. CONCLUSIONES

- Las ciudades inteligentes se caracterizan por simplificar los procesos de trabajo de variados sistemas, redes y servicios, mediante el IoT, gracias a la recopilación de datos en tiempo real, lo que permite un análisis eficiente de diferentes variables para la toma de decisiones. El IoT ofrece una amplia gama de tecnologías radio, que permiten la comunicación entre dispositivos, entre las cuales se destacan las LPWAN, al brindar una cobertura extensa a bajo costo de operación, larga vida de baterías y escalabilidad. El análisis de las tecnologías LPWAN realizado en el presente trabajo de grado, permitió definir la más conveniente para el desarrollo del piloto; bajo los requerimientos establecidos, con lo cual se determinó que LoRa cumple con las características respecto a cobertura, costo, frecuencia, escalabilidad, latencia, movilidad y transmisión continua de datos requeridos para dar funcionalidad al piloto de información al usuario.
- LoRa es una tecnología de Capa Física que usa parámetros de configuración como son el Factor de Ensanchamiento, el Ancho de Banda y la Tasa de codificación, los cuales impactan directamente en la cobertura del sistema, al determinar la sensibilidad del receptor; la velocidad de transmisión de datos y la robustez de la red frente a interferencias. De acuerdo a lo anterior para el piloto de información al usuario, se determina que la mejor configuración es de SF=7, BW=125KHZ y CR de 4/5, la cual proporciona un sistema con cobertura media, apta para cubrir el escenario establecido y una velocidad de transmisión de datos aceptable, recogiendo los datos necesarios para realizar el análisis de resultados de tiempo de espera del presente trabajo de grado.

- El piloto de información al usuario desarrollado en el trabajo de grado, se orienta a las Ciudades Inteligentes, las cuales buscan que el Sistema de Transporte Público sea el medio de transporte más utilizado por sus habitantes, esto se logra ofreciendo al usuario un servicio eficiente y confiable, facilitando la cotidianidad de las personas; basándose en dicha necesidad, el piloto es capaz de brindar información de tiempo de espera del bus en parada, con datos muy aproximados al tiempo real y su respectivo número de ruta, de igual forma se visualiza, sobre una página Web, la ubicación del bus, lo cual le permite al usuario verificarla en cada instante de tiempo
- El desempeño del piloto se determina principalmente mediante la prueba de confiabilidad, obteniendo que la variación entre el tiempo de espera desplegado por el piloto, el tiempo de espera medido por cronómetro y el tiempo de espera teórico, fue próximo a 4 segundos; se consideraron posibles inconsistencias en el cálculo de tiempo de espera del piloto, como el margen de error del GPS LoRa y la velocidad variable en la que se desplazó el MaB, generando un margen de error de 4 segundos, este valor no es crítico para una Sistema de Transporte Publico Inteligente real por lo tanto, se considera que el desempeño del piloto de información al usuario es satisfactorio.
- Las pruebas experimentales de cobertura, permitieron determinar un área de cobertura próxima a la cobertura estipulada en la teoría de la tecnología LoRa, alcanzando los dos kilómetros para entornos urbanos, se considera que la cobertura proporcionada por el dispositivo LoRa puede ser mayor si se ofrece línea de vista. La ubicación establecida para el MG proporciona la cobertura del escenario de pruebas, sin embargo, la zona de cobertura en general que puede brindar el dispositivo, está limitada por las obstrucciones como edificios y podría mejorarse si se ubica en un lugar más alto.

## 5.2. TRABAJOS FUTUROS

Los trabajos futuros están ligados a mejoras o adiciones más propicias para el diseño e implementación del sistema de información al usuario que se plantea en este trabajo.

1. Implementar mediante el uso de la tecnología LoRa un piloto para la transmisión de información al usuario en un sistema de control de flota,

permitiendo la coexistencia de dos o más dispositivos configurados en la misma frecuencia, evaluando las transmisiones ortogonales que proporciona el uso de diferentes factores de dispersión.

2. Implementar un piloto de transmisión de información al usuario en un sistema de control de flota, haciendo uso de otra tecnología LPWAN bajo parámetros de selección similares al presente trabajo, posteriormente hacer un análisis comparativo de la confiabilidad del tiempo de espera brindado por los pilotos ya desarrollados, determinando la tecnología más apropiada para este tipo de sistemas.
3. Agregar seguridad al piloto desarrollado por el presente trabajo, evaluando la capacidad del piloto contra usuarios mal intencionados.
4. Implementar mediante el uso de la tecnología LoRa un piloto para la transmisión de información al usuario en un sistema de control de flota, aumentando el número de buses y número de paradas en un escenario de implementación con mayor cobertura.

## Bibliografía

- [1] S. P. Mohanty, U. Choppali, and E. Kougianos, "Everything you wanted to know about smart cities," *IEEE Consumer Electronics Magazine*, pp. 60-70, July 2016.
- [2] L. L. Moreno Herrera and A. Gutierrez Sanchez, "Ciudades Inteligentes: Oportunidades para generar soluciones sostenibles," Cintel, Bogota, 2012.
- [3] Fundación Telefonica, "Smart Cities: El gran paso hacia el internet de las cosas," Fundación Telefonica, Madrid, España, 2011.
- [4] G. Corotinschi and V. Ghreorguita, "Smart cities become possible thanks to the internet of things," University of Suceava, Romania, 2015.
- [5] W. J. Mitchell, "Ciudades inteligente," *revista sobre la sociedad del conocimiento*, pp. 3-11, Octubre 2007.
- [6] C. Harrison et al, "Foundations for Smarter Cities," *IBM Journal of Research and Development*, vol. 54, no. 4, 2010.
- [7] L. Anthopoulos, "Defining Smart City Architecture for Sustainability," *Electronic Government and Electronic Participation*, pp. 140-147, 2015.
- [8] International Organization for Standardization. (2015, December) isotader.es. [Online]. <http://www.isotader.es/iso-37120-la-norma-de-las-ciudades-inteligentes/>
- [9] ITU. (2015) itu.int. [Online]. <https://www.itu.int/en/ITU-T/focusgroups/ssc/Pages/default.aspx>
- [10] Y. Xinping et al, "Research and Development of Intelligent Transportation Systems," *11 th International Symposium on Distributed Computing and Applications to Bussiness, Engineering & Science*, pp. 321-327, 2012.
- [11] M. Alam et al, "Introduction to Intelligent Transportation Systems," *Springer International Publishing Switzerland*, 2016.
- [12] J. Houghton, J. Reiners, and C. Lim, "Transporte inteligente," IBM, New york, 2009.
- [13] Avante. (2017, julio) avante.gov.co. [Online]. <https://www.avante.gov.co/operaciones/sistema-gestion-control-de-flota>
- [14] Smart Cities Atlas. (2016, june) smartcitiesatlas.com. [Online]. <http://www.smartcitiesatlas.com/el-control-de-flota-en-las-smart-cities/>
- [15] Smart Cities Atlas. (2016, may) smartcitiesatlas.com. [Online]. <http://www.smartcitiesatlas.com/transporte-y-movilidad-en-una-ciudad-inteligente/>
- [16] A. Salvi, M. Tufo, and E. Natale, "A fleet management service for smart cities:

- the S2-Move Project," 2016.
- [17] J. Basterretche, "Dispositivos Móviles," Universidad Nacional del Nordeste, Corrientes-Argentina, Trabajo de adscripción 2007.
- [18] J. C. Tapia Lara, "Conceptos básicos de telefonía celular," 2016.
- [19] J.J. Salamanca, L. Zambrano, and S. Malobabic, "Radio trunking," Universidad Simon Bolivar, 2004.
- [20] J. F. Monserrat del Rio, "Sistemas de telecomunicación privados," Escuela politecnica superior de Gandia,.
- [21] J. L. Peñarredonda, "La tecnología tras el metro de medellin y sus 750000 viajes diarios," *Enter.co*, mayo 2014.
- [22] W.Castro Plazas, M. Medina Palacio, and H. Montaña Quintero, "Sistemas Troncalizados Tetra," 2016.
- [23] Avante, "Tetra, una red para el sistema de transporte publico para la ciudad de Pasto," Avante, Pasto, 2016.
- [24] Agencia Nacional del Espectro, Resolucion N° 000711, 2016, Establece las bandas de frecuencias de libre utilización dentro del territorio nacional.
- [25] U. Raza, P. Kulkami, and M. Sooriyabandara, "Lower Power Wide Area Networks: An Overview," IEEE Communications Society, 2017.
- [26] D. Ismail, M. Rahman, and A. Saifullah, "Low-Power Wide-Area Networks: Opportunities, Challenges and Directions," in *Actas del Taller Programa de la XIX Conferencia Internacional sobre Computación Distribuida y Redes*, Varanasi, 2018.
- [27] E. Poorter, J. Hoebeke, and M. Strobbe, "Sub-GHz LPWAN Network Coexistence, Management," *Wireless Personal Communications*, vol. 95, pp. 187-213, July 2017.
- [28] P. Pardal, "Redes de Área Extensa para aplicaciones de IoT: modelado de comunicaciones Sigfox," Universidad Politecnica de Valencia, Tesis de master 2017.
- [29] S. Kheng Tan, M. Sooriyabandara, and Z. Fan, "M2M Communications in the Smart Grid: Applications, Standards, Enabling Technologies, and Research Changes," *International journal of Digital Multimedia Broadcasting*, Mayo 2011.
- [30] K. Pothugantil and A. Chitneni, "A Comparative Study of Wireless Protocols: Bluetooth,UWB, ZigBee, and WiFi," *Research India Publications*, vol. 4, no. 6, pp. 655-662, 2014.
- [31] Ubiquiti Networks. (2014) dl.ubnt.com. [Online].  
[https://dl.ubnt.com/datasheets/airmax/UBNT\\_DS\\_airMAX\\_TDMA.pdf](https://dl.ubnt.com/datasheets/airmax/UBNT_DS_airMAX_TDMA.pdf)
- [32] R. Kapur. (2018, february) everythingrf.com. [Online].

<https://www.everythingrf.com/community/what-is-rpma-technology>

- [33] J. Finnegan and S. Brown, "A Comparative Survey of LPWA Networking," National University of Ireland, February 2018.
- [34] Ingenu, "How RPMA works," Ingenu, Educational Guide 2018.
- [35] Waviot. (2018) waviot.com. [Online]. <http://waviot.com/technology/what-is-nb-fi>
- [36] Waviot, "Waviot NB-Fi LPWAN Technology," Waviot, 2016.
- [37] Telensa. (2018) telensa.com. [Online]. <https://www.telensa.com/technology#unb-network>
- [38] Telensa. (2018) telensa.com. [Online]. <https://www.telensa.com/>
- [39] LoRa Alliance. (2019) LoRa Alliance. [Online]. <https://lora-alliance.org/>
- [40] K. G. Beutel, "Power Utility Remote Device Communications using a Low Power Wide Area Network (LPWAN) Based on the LoRa Communications Standard," 2016.
- [41] LoRa Alliance, A technical overview of LoRa and LoRaWAN, 2015.
- [42] DASH7 Alliance. (2019) DASH7 Alliance. [Online]. <http://www.dash7-alliance.org/why-dash7/>
- [43] U. Mehboob et al., Survey of IoT Communication Protocols Techniques, Applications and Issues, 2016.
- [44] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low Power Wide Area Networks: An Overview," 2017.
- [45] D. Ismaili, M. Rahman, and A. Saifullah, "Low-Power Wide-Area Networks: Opportunities, Challenges, and Directions," Enero 2018.
- [46] E. Pooter et al., "Sub-GHz LPWAN Network Coexistence, Management and Virtualization: An Overview and Open Research Changes," *Springer Science*, junio 2017.
- [47] A. Oluwafemi, Design and Analysis of LTE and Wi-Fi Schemes for Communications of Massive Machine devices, 2017.
- [48] SigFox, SigFox Technical Overview, 2017.
- [49] A. Ali et al., "Technologies and Challenges in developing Machine-to-Machine applications: A survey," *Journal of Network and Computer Applications*, pp. 124-139, 2017.
- [50] Wisun Alliance. (2018) wi-sun.org. [Online]. <https://www.wi-sun.org/>
- [51] C. S. Sum, M. Rahman, and L. Lu, "On communication and interference range of IEEE 802.15.4g smart utility networks," in *Wireless Communications and Networking Conference (WCNC), 2012 IEEE*, Shanghai, 2012.
- [52] Semtech, AN1200.22 LoRa Modulation Basics, 2015.

- [53] C. Jansen et al., "Known and Unknown Facts of LoRa: Experiences from a Large-scale Measurement Study," 2019.
- [54] Semtech, SX1272/3/6/7/8: LoRa Modem Design Guide, 2013.
- [55] A. Augustin et al., "A Study of LoRa: Long Range & Low Power Networks for the Internet of Things," septiembre 2016.
- [56] Copyright © 2019 LoRa Alliance™. (2019) LoRa Alliance. [Online]. <https://lora-alliance.org/about-lorawan>
- [57] Copyright © 2019 LoRa Alliance™. (2015) LoRa Alliance. [Online]. <https://lora-alliance.org/sites/default/files/2018-04/what-is-lorawan.pdf>
- [58] Orange TM and Actility, LoRa Device Developer Guide, 2016.
- [59] Gemalto; Actility; Semtech, LoRaWAN security Full end-to-end encryption for IoT application providers, 2017.
- [60] LoRa Alliance, LoRaWAN 1.1 Regional Parameters, 2017.
- [61] Copyright © 2019 LoRa Alliance™. (2017) LoRaWAN 1.1. Regional Parameters. [Online]. [https://lora-alliance.org/sites/default/files/2018-04/lorawantm\\_regional\\_parameters\\_v1.1rb\\_-\\_final.pdf](https://lora-alliance.org/sites/default/files/2018-04/lorawantm_regional_parameters_v1.1rb_-_final.pdf)
- [62] Dragino. (2019) LoRa Questions. [Online]. [https://wiki.dragino.com/index.php?title=LoRa\\_Questions](https://wiki.dragino.com/index.php?title=LoRa_Questions)
- [63] Dragino Technology. (2017) Dragino. [Online]. <http://www.dragino.com/products/lora/item/108-lora-gps-shield.html>
- [64] Arduino. (2019) Arduino. [Online]. <https://www.arduino.cc/en/Main/arduinoBoardUno/>
- [65] Electrodragon. (2013) Electrodragon. [Online]. <https://www.electrodragon.com/product/mifare-rc522-rfid-card-readerdetector-ic-card/>
- [66] Dragino. (2017, octubre) Wiki Dragino. [Online]. [https://wiki.dragino.com/index.php?title=Lora\\_Shield](https://wiki.dragino.com/index.php?title=Lora_Shield)
- [67] Dragino Technology. (2019) Dragino. [Online]. <http://www.dragino.com/products/lora/item/118-olq01.html>
- [68] Dragino Technology. (2019) Dragino. [Online]. <http://www.dragino.com/products/module/item/102-lora-shield.html>
- [69] Arduino. (2019) Arduino. [Online]. <https://www.arduino.cc/en/main/arduinoBoardMega2560?setlang=cn>
- [70] Sunfonder. (2018) Sunfonder. [Online]. [http://wiki.sunfounder.cc/index.php?title=LCD1602\\_Module](http://wiki.sunfounder.cc/index.php?title=LCD1602_Module)
- [71] MathWorks. (2019) MathWorks. [Online]. <https://es.mathworks.com/help/thingspeak/?requestedDomain=>

- [72] Mike McCaulen. (2019) RadioHead. [Online]. <https://www.airspayce.com/mikem/arduino/RadioHead/index.html>
- [73] Mikal Hart. (2019) Arduiniana. [Online]. <http://arduiniana.org/libraries/tinygpsplus/>
- [74] Augusto Bernales. datospdf.com. [Online]. [https://datospdf.com/download/confiabilidad-5a44d161b7d7bc422b904023\\_pdf](https://datospdf.com/download/confiabilidad-5a44d161b7d7bc422b904023_pdf)
- [75] J. S. Rodriguez, "Antena helicoidal cuadrifilar de banda ancha para instrumentación," Pontificia Universidad Javeriana, Bogotá, 2013.
- [76] karlhennermann. (2009) karlhennermann.wordpress.com. [Online]. <https://karlhennermann.wordpress.com/2009/06/29/magellan-mobilemapper-6-gps-part-1/>
- [77] Xirio. xirio-online.com. [Online]. <https://www.xirio-online.com/help/es/index.htm>
- [78] A. Figuero Negrete, "Desarrollo de una herramienta de software para el diseño de aeropuertos basada en el anexo 14," Instituto Politecnico Nacional, Mexico, 2012.
- [79] Movilidad Futura SAS. (2018) movilidadfutura.gov.co. [Online]. <http://movilidadfutura.gov.co/>
- [80] Arduino. (2018) arduino.cc. [Online]. <https://www.arduino.cc/en/Guide/Introduction>
- [81] Dragino. (2018) wiki.dragino.com. [Online]. [http://wiki.dragino.com/index.php?title=Lora/GPS\\_Shield](http://wiki.dragino.com/index.php?title=Lora/GPS_Shield)
- [82] D. A. Rincon Lopez, R. Hernandez Jojoa, and J. Alvarez. (2011) es.slideshare.net. [Online]. <https://es.slideshare.net/davandrinlop/antenas-helicoidales-8654948>
- [83] Union Internacional de Telecomunicaciones, "Rec UIT-R P.526-14," Union Internacional de Telecomunicaciones, 2018.
- [84] Ruben Hernandez Jojoa, Jaime Alvarez David Andres Rincon Lopez. (2011) es.slideshare.net. [Online]. <https://es.slideshare.net/davandrinlop/antenas-helicoidales-8654948>

## APENDICE A: CONVERSIÓN DE COORDENADAS GEOGRÁFICAS A CARTESIANAS UTM

La conversión de coordenadas geográficas a cartesianas UTM se basa en las fórmulas de Coticchia-Surace. Dichas formulas fueron planteadas por Alberto Cotticia y Luciano Surace en el “Bolletino di Geodesia e Science Affini”. La precisión que se obtiene, es aproximadamente de un centímetro cuando se utilizan suficientes decimales [78].

### 1. Sobre la geometría del elipsoide

Inicialmente se calcula la excentricidad, la segunda excentricidad y el radio polar de curvatura, como se muestra en las siguientes ecuaciones:

- Excentricidad

$$e = \frac{\sqrt{a^2 - b^2}}{a}$$

- Segunda excentricidad

$$e' = \frac{\sqrt{a^2 - b^2}}{b}$$

- Radio polar de la tierra

$$c = \frac{a^2}{b}$$

Dónde: Semieje mayor(a): 6378388.0

Semieje menor (b): 6356911.946130

### 2. Sobre la longitud y la latitud

La latitud y la longitud que están definidas en grados minutos y segundos se convierten a grados decimales y posteriormente a radianes así:

- $Grados\ decimales = grados + \frac{minutos}{60} + \frac{segundos}{3600}$

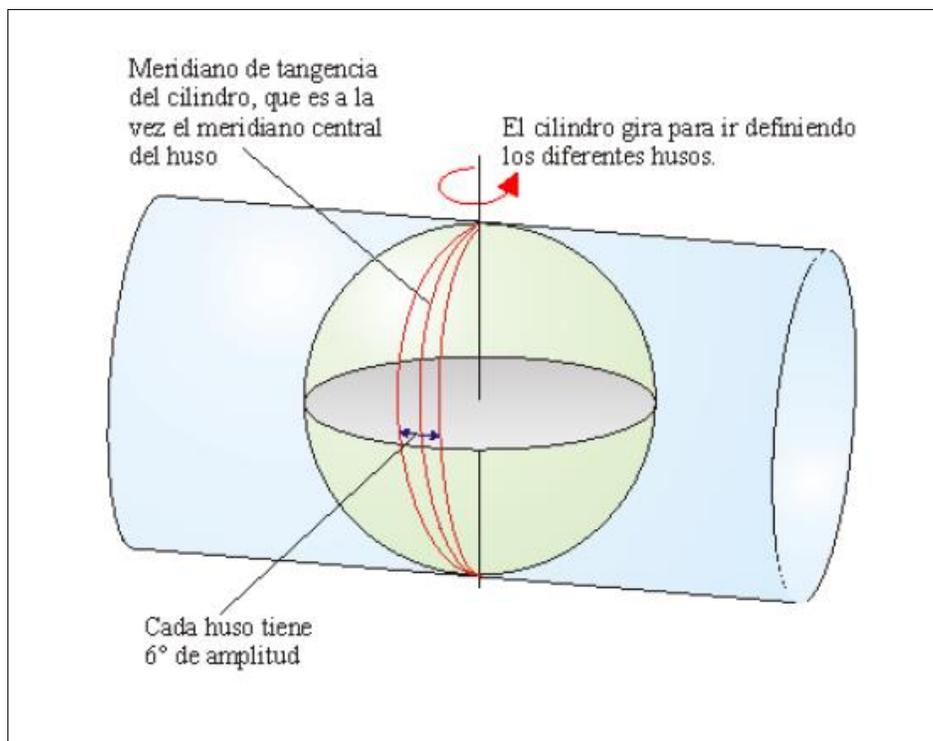
- $Radianes = \frac{Grados\ decimales * \pi}{180}$

Notación de las variables:

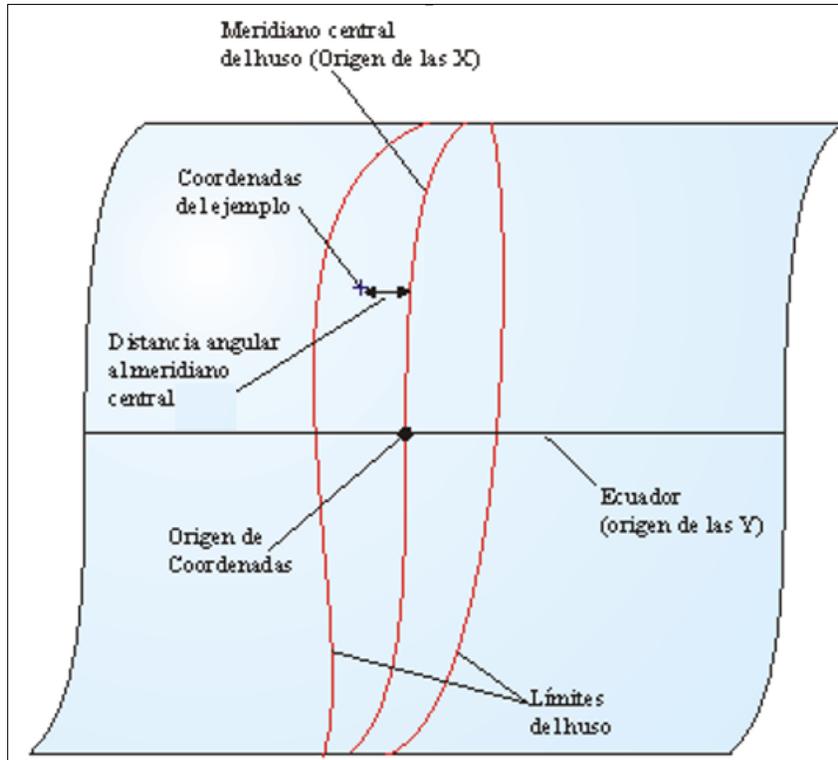
- Longitud en grados decimales: Xgd
- Latitud en grados decimales: Ygd
- Longitud en Radianes: x
- Latitud en Radianes: y

### 3. Sobre el huso horario

En la proyección UTM se utiliza el cilindro transverso como superficie desarrollable, este se gira virtualmente para definir los diferentes husos que rodean la tierra. Una vez conocidos los datos de longitud y latitud se calcula el huso o zona UTM, en la cual se encuentran las coordenadas a convertir. El cilindro transverso se observa en la *Figura A.1 y A.2*



**Figura A.1 Cilindro transverso**



**Figura A. 2 Cilindro transverso 2**

Para evitar las coordenadas X negativas al lado izquierdo del meridiano central se suma a las coordenadas X la cantidad de 500000, y para evitar las coordenadas negativas del eje Y, que se encuentren al sur del Ecuador se suma el valor de 10000000

Se procede a determinar el huso o zona UTM

- $Huso = \text{entero de } \left[ \frac{X_{gd}}{6} + 31 \right]$

Posteriormente se encuentra el meridiano central del huso.

- $x_0 = (Huso * 6) - 183$

Se calcula la distancia angular que existe entre la longitud del punto y el meridiano central del huso.

- $\Delta x = x - \frac{x_0 * \pi}{180}$

**4. Fórmulas Coticchia-Surace para convertir las coordenadas geográficas a cartesianas UTM**

- $A = \cos y * \text{sen } \Delta x$
- $E = \frac{1}{2} * \ln \frac{1+A}{1-A}$
- $\eta = \arctan \left[ \frac{\tan y}{\cos \Delta x} \right] - y$
- $v = \frac{c}{(1+e'^2 * \cos^2 y)^{\frac{1}{2}}} * 0.9996$
- $\epsilon = \frac{e'^2}{2} * E^2 * \cos^2 y$
- $A_1 = \text{sen}(2y)$
- $A_2 = A_1 * \cos^2 y$
- $J_2 = y + \frac{A_1}{2}$
- $J_4 = \frac{3 * J_2 + A_2}{4}$
- $J_6 = \frac{5 * J_4 + A_2 * \cos^2 y}{3}$
- $\alpha = \frac{3}{4} * e'^2$
- $\beta = \frac{5}{3} * \alpha^2$
- $\gamma = \frac{35}{27} * \alpha^3$
- $B_0 = 0.9996 * c * (y - \alpha * J_2 + \beta * J_4 - \gamma * J_6)$
- $X_{UTM} = E * v * \left( 1 + \frac{\epsilon}{3} \right) + 500000$
- $Y_{UTM} = \eta * v * (1 + \epsilon) + B_0$

Si la latitud se encuentra en el hemisferio sur del Ecuador se suma el valor de 10000000 así:

- $Y_{UTM} = \eta * v * (1 + \epsilon) + B_0 + 10000000$

## APENDICE B: ACCESO A LA GATEWAY OLG01 E INSTALACIÓN DE LA PLACA EN SOFTWARE ARDUINO

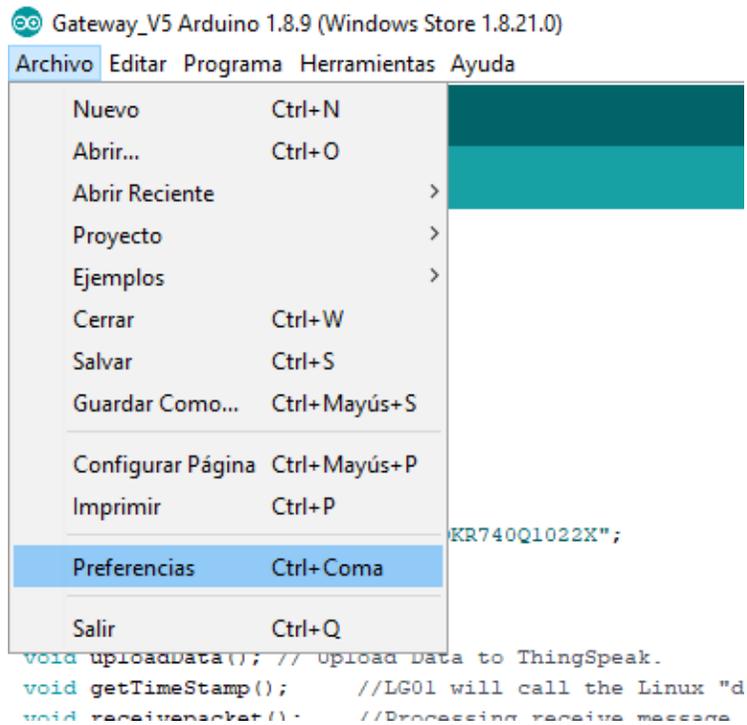
Para acceso a la red generada por la Gateway OLG01 e instalación de la placa en el entorno de desarrollo Arduino se debe seguir los pasos mencionados a continuación.

1. Para acceder a la red el desarrollador lo puede hacer por un punto de acceso WiFi generado por la Gateway o por un puerto Ethernet; sí se conecta por puerto Ethernet no se requiere ningún otro tipo de acción ya que la conexión es automática a diferencia de la conexión por el punto de acceso WiFi, en donde se tendrá que seleccionar la red con el nombre de Dragino-XXX generado por la Gateway. En la *Figura B.1* se observa la red generada por la Gateway con conexión WiFi.



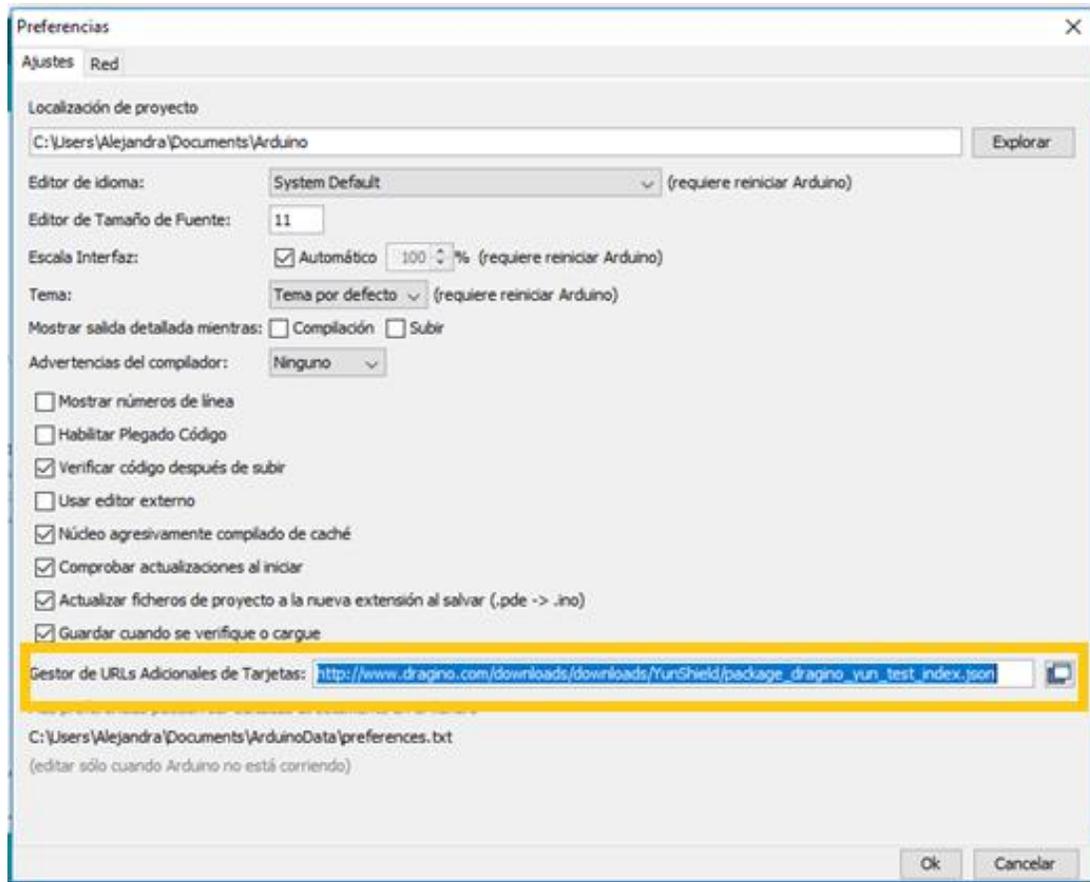
**Figura B. 1 Red Dragino-XXX generada por la Gateway OLG01.**

2. Una vez se selecciona y se conecta a la red, se procede hacer la instalación de la placa en el software de desarrollo Arduino.
3. En la interfaz de Arduino se selecciona archivo y posterior a esto se va a la casilla de referencias y se selecciona como se muestra en la *Figura. B.2*.



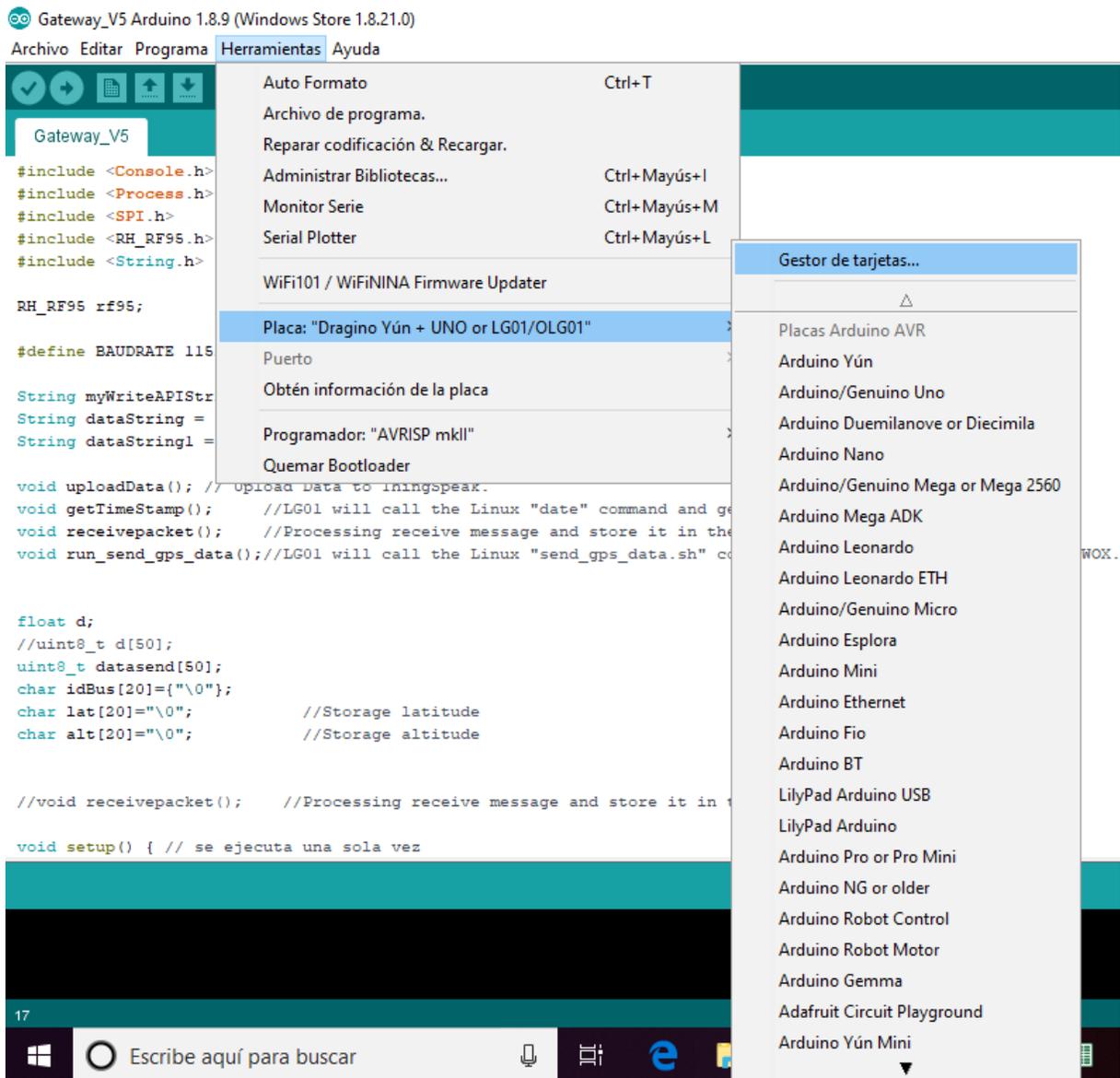
**Figura B.2 Paso a paso para llegar a la casilla preferencias en la interfaz de Arduino.**

4. Al seleccionar la casilla de preferencias se abre una interfaz como se muestra en la *Figura B.3* en donde se selecciona la pestaña de ajustes y en la casilla "Gestor de URLs adicionales de tarjetas:", se pega el siguiente link:  
[http://www.dragino.com/downloads/downloads/YunShield/package\\_dragino\\_yun\\_test\\_index.json](http://www.dragino.com/downloads/downloads/YunShield/package_dragino_yun_test_index.json); el cual es proporcionado por el manual de usuario de Gateway.

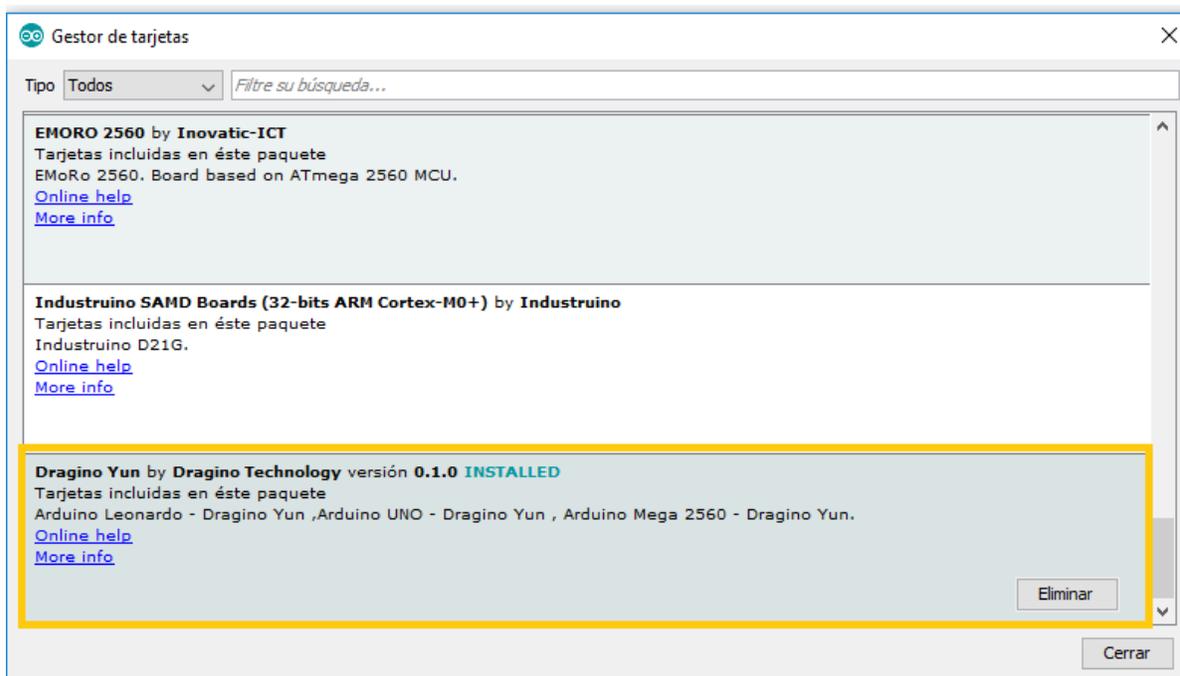


**Figura B. 3 Interfaz Preferencias en Arduino**

5. Una vez gestionada la URL de la tarjeta se procede hacer la instalación de esta seleccionando Herramientas – Placa – Gestor de Tarjetas como se muestra en la *Figura B.3*. En el momento de seleccionar Gestor de tarjetas se abre una interfaz y se busca la tarjeta a instalar en este caso *Dragino Yun by Dragino Technology*, y se selecciona Instalar como se observa en la *Figura B.4*.



**Figura B.4 Selección Gestor de Tarjetas en la interfaz de Arduino.**



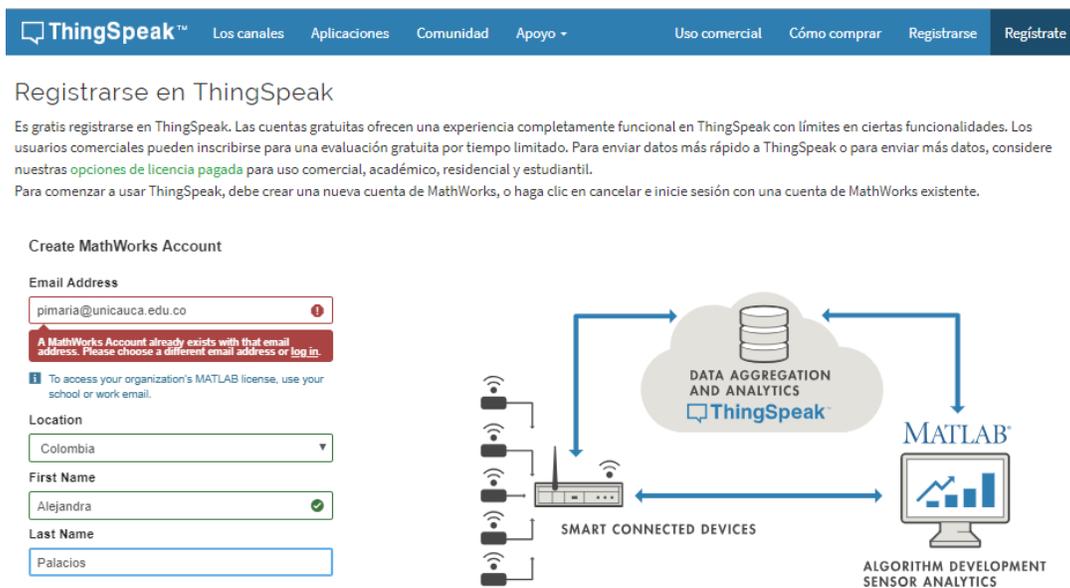
**Figura B. 5 Selección e instalación de la tarjeta Dragino Yun.**

**Nota:** El puerto generado por la tarjeta descrito anteriormente corresponde a la dirección IP: 10.130.1.1, el cual se usa para hacer la gestión de los datos del entorno de desarrollo de Arduino.

## APENDICE C: CREACIÓN DE CANALES Y OBTENCION DE LLAVES EN EL SERVIDOR THINGSPEAK.

A continuación, se detallan los pasos a seguir para el uso del servidor ThingSpeak, el cual proporciona canales y llaves para la gestión de datos.

1. Ingresando a la página de ThingSpeak, se selecciona la pestaña de registro y se proporciona un correo electrónico, localización y nombre de usuario, con el fin de tener acceso al servidor. En la *Figura C.1* se observa la interfaz en ThingSpeak para el registro.



The image shows the registration page of ThingSpeak. At the top is a navigation bar with links: 'Los canales', 'Aplicaciones', 'Comunidad', 'Apoyo', 'Uso comercial', 'Cómo comprar', 'Registrarse', and 'Regístrate'. The main heading is 'Registrarse en ThingSpeak'. Below it, there is a paragraph explaining that registration is free but some features are limited, and that users can opt for a paid license for commercial, academic, residential, or student use. It also states that a MathWorks account is required. Below the text is a 'Create MathWorks Account' form with fields for 'Email Address' (pimaria@unicauca.edu.co), 'Location' (Colombia), 'First Name' (Alejandra), and 'Last Name' (Palacios). A red error message indicates that a MathWorks account already exists with the provided email address. To the right of the form is a diagram illustrating the data flow: 'SMART CONNECTED DEVICES' send data to a cloud labeled 'DATA AGGREGATION AND ANALYTICS ThingSpeak', which then connects to a 'MATLAB' monitor labeled 'ALGORITHM DEVELOPMENT SENSOR ANALYTICS'.

**Figura C. 1** Interfaz servidor ThingSpeak para el registro de usuarios.

2. Posterior al registro se crea el canal seleccionando *New Chanel*, con un nombre, descripción y agregación de campos con un máximo de 8; en este caso el nombre del canal es Localización de buses con dos campos llamados latitud y longitud, realizado lo anterior se selecciona *Save Chanel*. En la *Figura C.2* se muestra la interfaz para creación del canal.

## My Channels

[New Channel](#)

Name	Created	Updated
Localización <a href="#">Private</a> <a href="#">Public</a> <a href="#">Settings</a> <a href="#">Sharing</a> <a href="#">API Keys</a> <a href="#">Data Import / Export</a>	2018-09-21	2018-10-30 21:52
Localización buses <a href="#">Private</a> <a href="#">Public</a> <a href="#">Settings</a> <a href="#">Sharing</a> <a href="#">API Keys</a> <a href="#">Data Import / Export</a>	2018-11-01	2018-11-01 15:56

## New Channel

Name

Description

Field 1

Field 2

Field 3

Field 4

Show Status

[Save Channel](#)

**Figura C.2** Interfaz de ThingSpeak para la creación de un canal.

- Para la obtención de las llaves seleccionamos la pestaña de *API Keys* como se muestra en la *Figura C.3.*, en donde se proporciona dos claves, una de escritura y otra de lectura para hacer gestión de datos como se puede ver en la *Figura C.4.*

## Localizacion buses

Channel ID: 616399  
Author: alepalacios031  
Access: Public

Private View Public View Channel Settings Sharing **API Keys** Data Import / Export

+ Add Visualizations + Add Widgets Export recent data

MATLAB Analysis MATLAB Visualization

### Channel Stats

Created: 5 months ago  
Last entry: about a month ago  
Entries: 2064

**Figura C. 3 Selección de la pestaña API Keys en la interfaz de ThingSpeak**

The screenshot shows the ThingSpeak interface with the 'API Keys' tab selected. The page is titled 'Localizacion buses' and shows channel information: Channel ID: 616399, Author: alepalacios031, Access: Public. The 'API Keys' tab is highlighted in the navigation bar. Below the navigation bar, there are two main sections: 'Write API Key' and 'Read API Keys'. The 'Write API Key' section has a text input field containing 'KTM1DKR740Q1022X' and a 'Generate New Write API Key' button. The 'Read API Keys' section has a text input field containing 'ZKLECB99POZ1VTR8', a 'Note' field, and buttons for 'Save Note' and 'Delete API Key'. Below these sections is a 'Generate New Read API Key' button. On the right side, there is a 'Help' section with a description of API keys and a list of 'API Keys Settings' and 'API Requests'. The 'API Requests' section shows three example requests: 'Update a Channel Feed', 'Get a Channel Feed', and 'Get a Channel Field', each with a corresponding URL and a 'Copy' button.

**Figura C.4 Claves proporcionadas por el servidor ThingSpeak**

## APENDICE D: OBTENCIÓN DE CLAVES SERVIDOR GOOGLE MAPS

Para hacer uso de los mapas proporcionados por el servidor de Google Maps se requiere tener acceso a las *API Key* de esto, a continuación, se muestra el proceso de la creación un proyecto en la plataforma de Google Cloud Platform para la obtención de la *API Key*.

1. De antemano con una cuenta de Google nos dirigimos al sitio web de Google Cloud Platform y creamos un proyecto, en este caso con el nombre de “sistema de localización” como se muestra en la *Figura D.1*.

Google Cloud Platform

Nuevo proyecto

Te quedan 23 projects en la cuota. Solicita un aumento o elimina proyectos. [Más información](#)

[MANAGE QUOTAS](#)

Nombre de proyecto \*  
Sistema de Localización

Algunos caracteres del nombre no son válidos. Escribe letras, números, comillas simples, guiones, espacios o signos de exclamación.

Organización  
unicauca.edu.co

Este proyecto se vinculará a unicauca.edu.co.

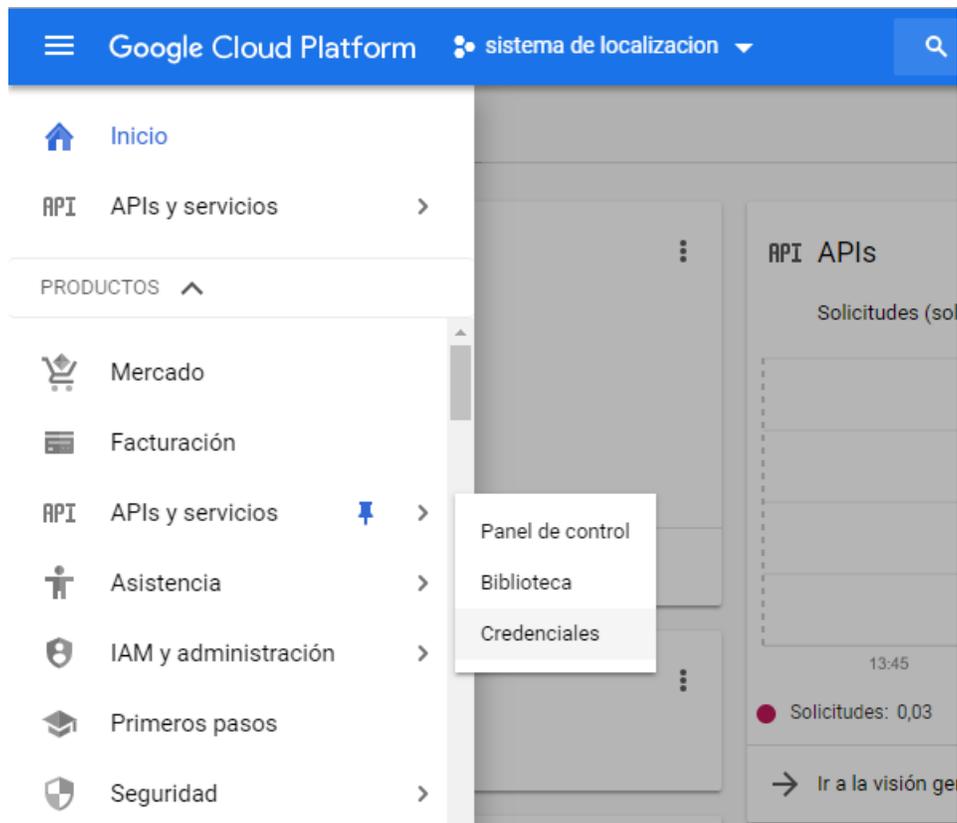
Ubicación \*  
unicauca.edu.co [EXPLORAR](#)

Carpeta u organización principal

[CREAR](#) [CANCELAR](#)

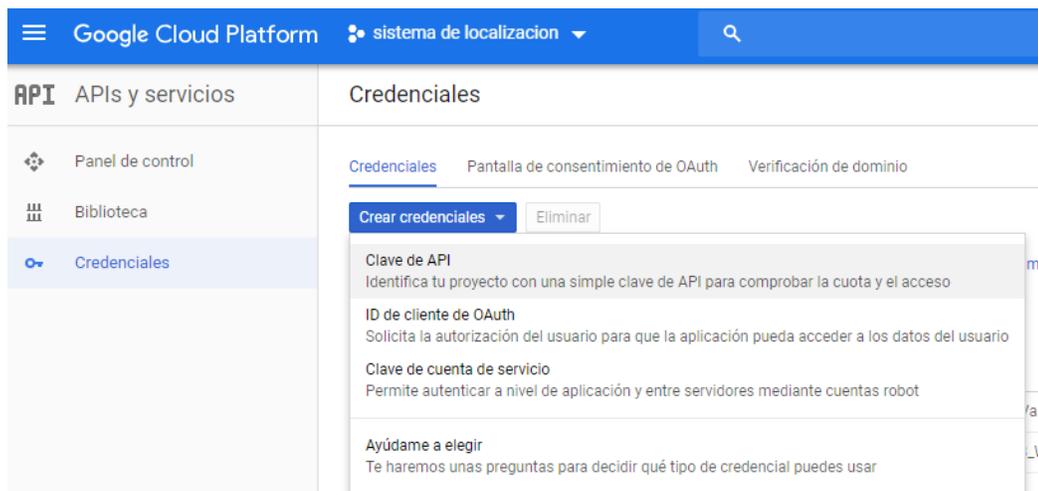
**Figura D.1 Creación de proyecto en Google Cloud Platform**

2. Una vez creada la cuenta nos dirigimos al menú de navegación y seleccionamos la pestaña APIs y servicios y seleccionamos credenciales como se evidencia en la *Figura D.2*.



**Figura D.2 Paso a paso para llegar a la pestaña de credenciales.**

3. Ya situados en la pestaña credenciales se procede a la selección de crear credenciales y a continuación Clave de API como se indica en la *Figura A.12*. Una vez ejecutado lo anterior se genera automáticamente una Clave API como se evidencia en la *Figura D.3*.



**Figura D.3 Paso a paso para la creación de claves APIs**

Google Cloud Platform sistema de localizacion

Restringir y cambiar el nombre a clave de API [VOLVER A GENERAR CLAVE](#) [ELIMINAR](#)

**⚠** This key is unrestricted.

Nombre \*  
API key

API Key  
AIzaSyDFuTD3Vaat-9wDgGRQn-sfb-Ub0drJoVE

Para usar esta clave en tu aplicación, transférela con el parámetro `key=CLAVE_API`.

**Restricciones de clave**  
Restrictions help prevent unauthorized use and quota theft. [Más información](#)

Fecha de creación	18 mar. 2019 9:43:29
Creada por	pimaria@unicauca.edu.co (tú)

**Figura D.4 Clave API Key ya creada.**

## APENDICE E: MANUAL DE USUARIO

Si se requiere implementar el piloto de información al usuario propuesto en el presente trabajo de grado para continuar con trabajos futuros, se deben realizar los siguientes pasos:

1. Extraer los archivos de la carpeta piloto de información al usuario con el nombre Gateway\_v5, correspondiente al MG; LoRa\_GPSRFID al MaB; primera parada, segunda parada, y tercera parada a las paradas del piloto y páginaweb. El programa en el cual pueden ser modificadas y ejecutadas las sentencias de código provenientes de los módulos MaB, MG y MP, es el software de desarrollo Arduino que puede ser descargado por medio del siguiente link <https://www.arduino.cc/en/Main/Software>; y para la página Web por medio de cualquier editor de texto como Notepad ++.
2. A continuación, se deben seguir los apéndices B, C y D con la finalidad de hacer la instalación de la placa de Dragino y acceso a la red generada por la Gateway OLG01, creación de canales sobre el servidor ThingSpeak y obtención de la API KEY de Google.
3. Siguiendo el diseño e implementación del piloto desarrollado en el presente documento, se hacen las conexiones hardware de los dispositivos radio utilizados.
4. Consolidadas las conexiones hardware y con disposición de los diferentes códigos del piloto, se procede a cargarlos sobre cada placa Arduino asignada a su correspondiente módulo.
5. Finalmente se obtiene el piloto en su total funcionalidad brindando información de tiempo de espera y número de ruta sobre cada LCD dispuesta en cada MP, además de la página Web que se puede observar simplemente abriendo mediante el navegador, el archivo páginaweb.