



POSGRADOS

MAESTRÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN

RPC-SO-19-No.277-2018

OPCIÓN DE
TITULACIÓN:

PROYECTO DE DESARROLLO

TEMA:

DESARROLLO DE UNA PLATAFORMA DE COMUNICACIÓN
BASADA EN IOT Y LPWAN PARA EL CONTROL Y
SUPERVISIÓN DE SISTEMAS DE ALUMBRADO PÚBLICO EN
PARQUES Y JARDINES.

AUTOR:

JOHNNY MARLON URDIALES AMAYA

DIRECTOR:

NINO TELLO VEGA URETA

GUAYAQUIL- ECUADOR
2022

Autor:



Johnny Marlon Urdiales Amaya.

Ingeniero en Electrónica en Control y Automatismo.
Candidato a Magíster en Electrónica y Automatización,
Mención en Informática Industrial por la Universidad
Politécnica Salesiana - Sede Guayaquil.
jurdialesa@est.ups.edu.ec

Dirigido por:



Nino Tello Vega Ureta.

Ingeniero en Electricidad Especialización Industrial
Magister en Automatización y Control Industrial.
nvega@ups.edu.ec

Todos los derechos reservados.

Queda prohibida, salvo excepción prevista en la Ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra para fines comerciales, sin contar con la autorización de los titulares de propiedad intelectual. La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual. Se permite la libre difusión de este texto con fines académicos e investigativos por cualquier medio, con la debida notificación a los autores.

DERECHOS RESERVADOS

©2022 Universidad Politécnica Salesiana.

GUAYAQUIL – ECUADOR – SUDAMÉRICA

JOHNNY MARLON URDIALES AMAYA.

DESARROLLO DE UNA PLATAFORMA DE COMUNICACIÓN BASADA EN
IOT Y LPWAN PARA EL CONTROL Y SUPERVISIÓN DE SISTEMAS DE
ALUMBRADO PÚBLICO EN PARQUES Y JARDINES.

Índice general

Índice de Figuras	6
Índice de Tablas	10
Resumen	11
Abstract	11
1. Introducción	1
1.1. Descripción general del problema	3
1.2. Objetivos	4
1.2.1. Objetivo general	4
1.2.2. Objetivos específicos	4
1.3. Contribuciones	5
1.4. Organización del manuscrito	6
1.5. Formulación del Problema	7
2. Marco Teórico	8
2.1. Estado del Arte	9
2.2. Definiciones Previas	13
2.2.1. Arduino	13
2.2.2. Sensores	16
2.2.3. Tecnología IOT	19
2.2.4. Aplicaciones	22
2.2.5. Tecnología LORA	24
2.2.6. LORAWAN	27
2.2.7. LPWAN	32
2.2.8. Plataforma TagoIO	33
2.2.9. Controlador Fuzzy	35

3. Marco Metodológico	37
3.1. Diseño del experimento	38
3.1.1. Diseño de la Red LoRa	38
3.1.2. Diseño del Dashboard	45
3.1.3. Diseño del Fuzzy	56
3.1.4. Diseño de la tarjeta PCB	65
3.2. Implementación del proyecto	66
3.2.1. Implementación de la red LoRa	66
3.2.2. Implementación de Nextion HMI	77
3.2.3. Implementación del controlador Fuzzy	84
3.2.4. Implementación de la placa PCB	85
3.3. Prueba del proyecto	87
3.3.1. Prueba del controlador Fuzzy	87
4. Resultados y Discusión	91
5. Conclusiones y Recomendaciones	94
5.1. Conclusiones	94
5.2. Recomendaciones	96

Índice de Figuras

2.1. Arduino Uno. Fuente: Store [2019]	14
2.2. Arduino LilyPAD. Fuente: Lilypad [2019]	14
2.3. Arduino Mega. Fuente: Store [2019]	15
2.4. Arduino Nano. Fuente: Store [2019]	15
2.5. Sensor de Corriente. Fuente: Naylmap [2018]	16
2.6. Módulo Luxómetro. Fuente: El autor	17
2.7. Sensor de Corriente. Fuente: Naylmap [2018]	18
2.8. Sensor de Movimiento. Fuente PCEL [2018]	19
2.9. Internet de las Cosas TIBCO [2021]	20
2.10. Componentes de un proyecto en IoT. Fuente: Romo-González and de los Ángeles Villalobos-Alonzo [2019]	22
2.11. Componentes de un proyecto en IoT. Fuente: ALFAIOT [2019]	25
2.12. Arquitectura de red LoRa. Fuente: Cárdenas et al. [2018]	27
2.13. Topología de estrella LoRaWAN. Fuente: CÁRDENAS [2018].	30
2.14. Topología punto a punto. Fuente: Rafael and Jiménez [2019].	31
2.15. Topología punto a punto. Fuente: Rafael and Jiménez [2019].	31
2.16. Nube TagoIO: Modelo de Aplicación. Fuente Lucas [2019]	34
2.17. Modelo de Fuzzy. Fuente: El autor	35
3.1. Diagrama de la red LoRa. Fuente: El autor	39
3.2. Sensor de Corriente. Fuente: El autor	40
3.3. Conexión del sensor de corriente. Fuente: El autor	40
3.4. Resultado de la medición de corriente. Fuente: Naylmap [2018]	41
3.5. Módulo Luxómetro. Fuente: El autor	42
3.6. Módulo Luxómetro. Fuente: El autor	42
3.7. Sensor de Voltaje. Fuente: El autor	43
3.8. Conexión de Voltaje. Fuente: El autor	44
3.9. Sensor de Movimiento. Fuente: Systems [2022]	45
3.10. Receptor JFL. Fuente: El autor	45

3.11. Panel de selección para la vinculación del TTN con Tago. Fuente: El autor	46
3.12. Creación del proyecto en Tago IO. Fuente: El autor	47
3.13. Creación del token de Tago IO. Fuente: El autor	47
3.14. Generación del ID para vinculación al TTN. Fuente: El autor	48
3.15. Incorporación del dispositivo a la plataforma. Fuente: El autor	48
3.16. Inserción del id de Dragino en la plataforma. Fuente: El autor	49
3.17. Ruta para la generación del Token ID. Fuente: El autor	49
3.18. Live Inspector del Dispositivo Lora. Fuente: El autor	50
3.19. Selección de la opción de Tablero. Fuente: El autor	50
3.20. Selección del nombre y tipo del tablero. Fuente: El autor	51
3.21. Incorporación de Widgets. Fuente: El autor	51
3.22. Incorporación del Widget Solid. Fuente: El autor	52
3.23. Vinculación al Dispositivo agregado. Fuente: El autor	52
3.24. Personalización de colores. Fuente: El autor	53
3.25. Personalización de rangos de valores. Fuente: El autor	53
3.26. Incorporación del Widget Line. Fuente: El autor	54
3.27. Vinculación al Dispositivo agregado. Fuente: El autor	54
3.28. Widget Icon. Fuente: El autor	55
3.29. Personalización de Widget. Fuente: El autor	55
3.30. Selección de iconos. Fuente: El autor	56
3.31. Inserción de Condiciones. Fuente: El autor	56
3.32. Universo de discurso del error. Fuente: El autor	56
3.33. Universo de discurso del PWM. Fuente: El autor	57
3.34. Universo de discurso del PWM. Fuente: El autor	57
3.35. Proceso de Defusicación. Fuente: El autor	58
3.36. Diagrama de conexión. Fuente: El autor	58
3.37. Diagrama de conexión. Fuente: El autor	59
3.38. Controlador del Fuzzy. Fuente: El autor	59
3.39. Funciones de membresía. Fuente: El autor	60
3.40. Funciones de membresía. Fuente: El autor	61
3.41. Funciones de membresía. Fuente: El autor	61
3.42. Controlador Fuzzy dentro de Simulink. Fuente: El autor	62
3.43. Señal de control. Fuente: El autor	62
3.44. Señal de control. Fuente: El autor	62
3.45. Señal de control. Fuente: El autor	63
3.46. Señal de control. Fuente: El autor	63
3.47. Señal de control. Fuente: El autor	63
3.48. Señal de control. Fuente: El autor	64
3.49. Señal de control. Fuente: El autor	64

3.50. Señal de control. Fuente: El autor	64
3.51. Diagrama del circuito. Fuente: El autor	65
3.52. Diagrama del circuito controlador de la lámpara. Fuente: El autor	66
3.53. Conversor analógico digital del microcontrolador. Fuente: El autor	66
3.54. Gateway LG308. Fuente: El autor	67
3.55. Conexión a través de WiFi Fuente: El autor	67
3.56. Conexión por WAN. Fuente: El autor	68
3.57. Ingreso a la plataforma. Fuente: El autor	68
3.58. Ingreso de las credenciales. Fuente: El autor	68
3.59. Configuración del WiFi Client. Fuente: El autor	69
3.60. Obtención del Gateway ID. Fuente: El autor	69
3.61. Selección del clúster. Fuente: El autor	70
3.62. Agregar el Gateway. Fuente: El autor	71
3.63. Inserción del ID. Fuente: El autor	71
3.64. Selección de frecuencia. Fuente: El autor	71
3.65. Selección del proveedor del servidor adecuado. Fuente: El autor	72
3.66. Verificación de la conexión LoRaWAN. Fuente: El autor	72
3.67. Agregar aplicación. Fuente: El autor	73
3.68. Registro del dispositivo final. Fuente: El autor	74
3.69. Modo de activación OTAA. Fuente: El autor	74
3.70. Ingreso del ID. Fuente: El autor	75
3.71. Conexión a la red. Fuente: El autor	75
3.72. Ingreso de tokens. Fuente: El autor	76
3.73. Códec de ingreso de keys. Fuente: El autor	76
3.74. Status del equipo. Fuente: El autor	77
3.75. Proceso para la conversión a decimal. Fuente: El autor	77
3.76. Creación de un nuevo archivo. Fuente: El autor	78
3.77. Elección de la plantilla. Fuente: El autor	78
3.78. Elección de la plantilla. Fuente: El autor	79
3.79. Nuevo proyecto en Inkscape. Fuente: El autor	79
3.80. Selección del fondo para el proyecto. Fuente: El autor	80
3.81. Dimensionar la pantalla. Fuente: El autor	80
3.82. Diseños realizados. Fuente: El autor	81
3.83. Proceso de exportación. Fuente: El autor	81
3.84. Agregar el diseño al editor. Fuente: El autor	82
3.85. Diseño de plantilla. Fuente: El autor	82
3.86. Sobre posición de cajas de textos. Fuente: El autor	83
3.87. Sobre posición de los botones. Fuente: El autor	83

3.88. Diagrama del circuito controlador de la lámpara. Fuente: El autor	86
3.89. Diagrama de la tarejta en doble cara. Fuente: El autor	86
3.90. Simulación del acabado de la tarjeta. Fuente: El autor	86
3.91. Circuito Impreso. Fuente: El autor	87
3.92. Señal de prueba de la entrada de la Lámpara. Fuente: El autor	88
3.93. Relación de lux con el tiempo. Fuente: El autor	88
3.94. Relación de lux con el tiempo. Fuente: El autor	89
3.95. Relación de lux con el tiempo. Fuente: El autor	89
3.96. Modelo gráfico de la planta real. Fuente: El autor	90
3.97. Modelo gráfico del modelo planteado. Fuente: El autor	90
3.98. Universo de discurso de entrada (error). Fuente: El autor	90
3.99. Universo de discurso de salida (PWM). Fuente: El autor	90
4.1. Comprobación del Fuzzy. Fuente: El autor	91
4.2. Comprobación del Fuzzy. Fuente: El autor	92
4.3. Comprobación del Fuzzy. Fuente: El autor	92
4.4. Comprobación del Fuzzy. Fuente: El autor	93
4.5. Comprobación del Fuzzy. Fuente: El autor	93

Índice de Tablas

2.1. Esquema general del modelo de referencia de Iot	21
2.2. Frecuencias por regiones	26
3.1. Esquema de conexión entre controladores y el Módulo BH1750	42
3.2. Tabla de Entrada y Salida	57
3.3. Tabla de extracción de variables	87

Resumen

En Guayaquil son escasos los sistemas automatizados que sean capaces de suministrar, optimizar, y tomar medidas de los parámetros del consumo eléctrico, que comunique a los supervisores en caso de que se presenten alguna situación anómala que afecte a los sistemas de alumbrado público. En este proyecto se plantea como objetivo, el desarrollo de una plataforma de comunicación basada en IoT y LPWAN que permita el control y supervisión de la iluminación en un parque. Como primer paso se analizó los tipos de tecnologías de la red LPWAN para los procesos de automatización basados en IoT, con el fin de obtener datos para este proyecto. En la implementación de este trabajo, se diseñó un interfaz web, en donde se puede visualizar y monitorear las variables como: Corriente voltaje y luxes en tiempo real, para así enviar a la plataforma TTN, y con el software Tago IO, se pueda observar los datos enviados por los sensores. Se optimiza la cantidad de luxes presentes en un parque, usando un sistema de control Fuzzy, para lo cual, fue necesario obtener el modelo matemático del sistema de una luminaria y con una herramienta computacional como Matlab, se calibró los parámetros de este controlador, los mismos que fueron colocados en el sistema embebido a través de un algoritmo, el cual permitió hacer eficiente la cantidad de luces enviadas a las diferentes áreas del parque. Como tarjeta electrónica controladora se tiene un sistema embebido, con el objetivo de abaratar costos. Durante el desarrollo de la parte electrónica fue necesario realizar dos circuitos de acoplamiento de señal, para los sensores y el control del PWM de la luminaria. Además, se realizó un circuito de estabilización a 5v para alimentar al sistema embebidos y al módulo de Lora. Al final del presente proyecto se evaluó la funcionalidad y viabilidad de la aplicación del sistema, en base a la toma diaria de los datos eléctricos, los cuales se generan según los valores establecidos durante la calibración. Una de las ventajas de este trabajo es el uso de la tecnología IoT, que permitió gestionar los datos capturados en tiempo real.

Abstract

In Guayaquil, there are few automated systems that are capable of supplying, optimizing, and taking measurements of the parameters of electrical consumption, which communicate to the supervisors in the event of any anomalous situation that affects the public lighting systems. The objective of this project is the development of a communication platform based on IoT and LPWAN that allows the control and supervision of lighting in a park. As a first step we analyzed the types of LPWAN network technology for automation processes based on IoT, in order to obtain data for this project, in the implementation of this work, a web interface was designed, where you can view and monitor the variables such as current, voltage and luxes in real time, in order to send it to the TTN platform, and with the Tago IO software, you can observe the data sent by the sensors. The amount of luxes present in a park is optimized using a Fuzzy control system, for which it was necessary to obtain the mathematical model of the system of a luminaire, and with a computational tool such as MatLab, the parameters of this controller were calibrated, which were placed in the embedded system through an algorithm, which allowed to make efficient the amount of lights sent to the different areas of the park. The controller electronic card is an embedded system, in order to reduce costs. During the development of the electronic part, it was necessary to couple the signals from the sensors and the one received by the embedded system, as well as the one sent to the power stage through signal amplification. With this procedure, the lighting system could be regulated in an automated way, working with the current, voltage and lux values established according to the tests carried out in the park. At the end of this project, the functionality and feasibility of the application of the system was evaluated, based on the daily collection of electrical data, which were generated according to the values established during the calibration, one of the advantages of this work is the use of IoT technology, which allowed managing the data captured in real time.

Capítulo 1

Introducción

Los sistemas de supervisión pueden ser manipulados con equipos a distancia por medio de las plataformas IoT con las cuales se obtienen datos sobre variables eléctricas, como la tensión y las corrientes de funcionamiento en tiempo real, a través de los equipos o sensores interconectados, lo que facilita administrar datos que pueden servir de referencia para una toma de decisiones eficaces en caso de un posible fallo interno del proceso.

Las instalaciones de alumbrado público se utilizan de forma continua según los horarios establecidos por las autoridades reguladoras. Este funcionamiento conlleva un consumo excesivo de electricidad el cual produce una degradación del rendimiento con un aumento de mantenimiento de las instalaciones lo cual requiere una supervisión periódica por parte de la empresa responsable.

El parque de Guayaquil carece de un sistema de iluminación controlable, por lo que se debe tener un sistema que acceda a la información relevante para el mantenimiento preventivo de la iluminación LED existente, sino que también reduce el consumo de energía teniendo como resultado un aumento de vida útil y el coste de las luminarias a partir de esta problemática se propuso un sistema de iluminación inteligente utilizando dispositivos periféricos electrónicos del cual se destaca Lora WAN, que se caracteriza por permitir la conexión de dispositivos ubicados a grandes distancias, a un bajo consumo energético, convirtiéndose en una opción de inversión para el diseño de sistemas inalámbricos, lo que ha permitido trabajar con el Internet de las Cosas (IoT).

Considerado este escenario, se realizó un diseño e implementación de dispositivos y sensores de tipo LPWAN, direccionada especialmente al control del sistema de alumbrado público mediante una plataforma que

permite monitorear desde cualquier dispositivo electrónico sobre situaciones o eventos que afecten el normal desenvolvimiento de los parques y jardines en la ciudad de Guayaquil, recopilando información que permita tomar decisiones efectivas ante cualquier anomalía de funcionamiento.

1.1. Descripción general del problema

En el Ecuador, los sistemas de alumbrado público de carreteras y parques operan durante largos períodos de tiempo en horarios establecidos por los organismos regulatorios [006/18 \[2018\]](#) este tipo de operación obliga a que las luminarias generen un consumo eléctrico excesivo y un deterioro progresivo. Por otro lado, el mantenimiento de estas luminarias requiere un monitoreo de campo periódico por el lado de la empresa a cargo, que a su vez conlleva a un consumo de recursos y tiempo, que puede ser aprovechado en otro tipo de actividades enfocadas al mejoramiento del sistema de alumbrado. En los parques de Guayaquil no existe un sistema de iluminación que sea monitoreado y controlado, que permita, obtener información pertinente para realizar un mantenimiento predictivo y preventivo de toda la red de alumbrado con luminarias de tipo led ya existente, así como reducir el consumo de energía , aumentar la vida útil de luminarias y costos.[Majdi \[2017\]](#),[Ahmed \[2020\]](#), la tecnología de luminarias led reducirá el coste del sistema convencional en un 50-60 La interconexión a través de redes entre dispositivos y objetos (IoT) hace visibles los sensores y dispositivos, recoge los datos aportados por cada unidad, facilitando eficazmente la detección de fallos y la toma de decisiones basadas en eventos pasados.

1.2. Objetivos

1.2.1. Objetivo general

Desarrollo de una plataforma de comunicación basada en IOT y LPWAN para el control y supervisión de sistemas de alumbrado público en parques y jardines.

1.2.2. Objetivos específicos

- Analizar los tipos de tecnología de red LPWAN empleados en los procesos de automatización basados en el internet de las cosas (IoT), para luego su aplicación de la tecnología LPWAN en los sistemas de iluminación inteligente en parques.
- Diseñar una interfaz web, para visualizar, monitorear y almacenar los datos, del sistema de iluminación en una plataforma comercial que exista en el medio, usando tecnología Iot, que permita detectar fallas
- Desarrollar un sistema de control Fussy que permita controlar la cantidad de luxes enviadas a las diferentes áreas de un parque.
- Evaluar la funcionalidad y viabilidad de la aplicación del sistema, analizando las señales de control y la base de datos adquiridas desde el sitio de prueba en comparación con los datos enviados por los sensores al sistema web.

1.3. Contribuciones

La mayoría de los sistemas de vigilancia utilizados en el sector industrial son remotos y recogen datos sobre variables eléctricas, como voltaje y corriente de operación proporcionada por los equipos o sensores que se encuentran interconectados entre sí, esto facilita el proceso de recopilación de datos que sirven de referente para la toma de decisiones efectivas ante posibles fallas en los procesos internos [Chen et al. \[2019\]](#), [Series and Science \[2020\]](#). Mediante el control programado de sensores y dispositivos actuadores para ser aplicado en los parques de la ciudad de Guayaquil, nos permitirá obtener información pertinente para realizar un mantenimiento predictivo y preventivo de toda la red de alumbrado con luminarias de tipo led ya existente, es así, que con la tecnología de luminarias led, permitirá reducir el costo del sistema convencional en un 50-60 %, lo que mejora la economía del país y ahorra una gran cantidad de inversión [Anitha et al. \[2018\]](#), [Abinaya \[2017\]](#).

Conforme con la necesidad del usuario, se ofrecerá una comunicación bidireccional con los dispositivos Lora, a través del protocolo Lora WAN, siendo esta una opción altamente empleada en el campo de las IoT.

Una vez obtenido el modelo matemático para el control y monitoreo de las luminarias existentes en el parque, se desarrollará el controlador Fuzzy a colocar en la tecnología seleccionada para la implementación permitiendo disminuir tanto el consumo de energía como los tiempos de respuesta, además también aumentará la vida útil de las lámparas, manteniendo una iluminación eficiente en los parques de la urbe.

1.4. Organización del manuscrito

La tesis se compone por tres capítulos, cada uno con un contenido específico que proporciona al lector las herramientas y los conocimientos necesarios para comprender los conceptos correspondientes. También hay que señalar que el documento de investigación está estructurado de acuerdo con la estructura propuesta por la Universidad Politécnica Salesiana, que también se describe detalladamente a continuación:

- Capítulo 1: Introducción, Descripción del Problema, Ojetivo General, Objetivo Específico 1, Objetivo Específico 2, Objetivo Específico 3, Objetivo Específico 4.
- Capítulo 2: Marco Teórico, Estado del Arte, Definiciones previas, Formulación del Problema.
- Capítulo 3: Diseño del Experimento, Desarrollo, Resultados y Discusión.
- Finalmente se adjunta la Bibliografía correspondiente.

1.5. Formulación del Problema

El problema al cual se busca dar solución a través del planteamiento de esta propuesta básicamente se trata de mitigar la carencia de herramientas de supervisión del alumbrado público, tanto en los parques como en jardines. De este modo la falta de supervisión hace que el uso del consumo eléctrico sea excesivo, además de que sin supervisión adecuada las luminarias sufren un deterioro progresivo. No cabe duda de que la falta de herramientas de vigilancia del alumbrado es una de las principales razones que provocan la contaminación ambiental, la cual afecta a la salud de los habitantes de la ciudad de Guayaquil. Asimismo, para aclarar los límites del desarrollo del proyecto se realiza una delimitación del campo de investigación, los cuales se detallan a continuación:

- Delimitación geográfica.
- Delimitación del objeto de estudio.
- Sistema del alumbrado público de parques y jardines

Del mismo modo otra de las grandes problemáticas que han surgido a lo largo de los años en los parques y jardines de Guayaquil es el uso de lámparas de corta duración y el consumo innecesario de energía que contamina el medio ambiente por la excesiva y elevada cantidad de residuos eléctricos. Otra observación analizada es que el personal de mantenimiento son los responsables de revisar los interruptores de las luces para asegurarse de que funcionan correctamente, un proceso que lleva mucho tiempo y limita las funciones que podrían realizarse en otra área.

Capítulo 2

Marco Teórico

En esta sección se detalla el estado del arte, el cual incluye una investigación pertinente al tema de este proyecto. El objetivo es ofrecer contenidos que contribuyan al tema de las redes de sensores inalámbricos y conocer mejor la realidad de la iluminación de espacios públicos, como parques y jardines, así como la implantación de nuevas tecnologías inalámbricas como LoRaWAN e IoT. Toda la información presentada en este capítulo permite el desarrollo del conocimiento dentro de las nuevas tecnologías de redes inalámbricas, desarrollando nuevas características teóricas dando un impulso en la realización práctica de la misma. De la misma manera, en este capítulo se presentan conceptos y puntos de vistas que el autor realiza en base a la información realizada por otros autores. Para este capítulo también se detallan las definiciones que forman parte fundamental dentro de este trabajo cuyo objetivo es desarrollar una plataforma de comunicación para llevar un control y supervisión de sistemas de alumbrado público. Para la elaboración de este tema, es importante obtener conocimientos previos sobre lo que es IoT, estándares de red, arquitectura de red, estándares de iluminación, magnitudes que deben ser consideradas para la lectura y presentación de datos, entre otros. Y por medio de la formulación del presente problema, se desarrolla la investigación en donde permitirá alcanzar los límites propuestos para este proyecto y así brindar una posible solución ante este problema tan habitual que ocurre dentro de las diversas ciudades del país.

2.1. Estado del Arte

El autor [Bruno, 2017], cito a Kevin Ashton quien fue el que uso la terminología IoT, comento: Si las computadoras pudieran saber todo acerca de datos recogidos, permitirían un importante ahorro de costes, debido a que todo se describe detalladamente sin intervención humana, lo cual se tendrá conocimientos cuándo sea necesario reparar, cambiar o restaurar las cosas, incluso si son nuevas o están desactualizadas. En el caso de IoT, su potencial consiste en revolucionar el mundo, como lo hizo Internet.

En un estudio realizado en la Universidad Técnica de Valencia por Garcés [2017], se indica que, las personas de todo el mundo viven conectados al Internet de las Cosas (IoT), estos dispositivos se enlazan en las redes de datos existentes, LTE, Wi-Fi y Bluetooth. Así mismo indica que, cada día surgen nuevas tecnologías, entre las cuales es similar a las llamadas LPWAN (Low Power Wide Area Networks) así como Lora, Sigfox y RPMA. El estudio se centró en una revisión teórica basándose en la comunicación LPWAN, de lo cual el autor concluye que las características de Sigfox, permite que esta tecnología sea una de las más aclamadas en el campo, además de formar parte de varios tipos de dispositivos. Para finalizar en la implementación del modelo NS-3, el cual se hace posible estudiar una infinidad de características que se encuentran presentes en diferentes entornos, entre las cuales destacan la cobertura, el consumo de potencia de energía, o la tasa de errores.

Asimismo, en Madrid, Olleros Guerrero [2020], llevó a cabo el desarrollo de un sistema basado en tecnologías IoT y DTL, con la finalidad de poder garantizar la calidad a lo largo de la cadena de suministro, aplicado a transporte refrigerado. El proyecto se desarrolló mediante una descomposición de cuatro bloques principales, el hardware, tecnología de comunicación, el almacenamiento en la nube y la plataforma blockchain, así mismo, dentro del desarrollo del sistema, se hicieron uso de herramientas como Sigfox y Arduino, con lo cual se consiguió que la aplicación pudiera contar con conectividad en todas las plataformas. Finalmente, el conocimiento acerca de protocolos y tecnologías que giran en torno a IoT ha ido en aumento, permitiendo incluso que se tome en consideración su aplicación en líneas de investigación de proyectos futuros.

Dentro del mismo contexto, en Las Palmas de Gran Canaria en España, Loubany et al. [2020], llevaron a cabo un proyecto basado en el desarrollo de una plataforma HW/SW, aplicaciones IoT con tecnologías VLC, BLE y Lora/Lora WAN. Este sistema tuvo como finalidad mostrarse como una plataforma genérica, que permite la adaptación en diferentes aplicaciones de IoT, así mismo, como metodología empleada dentro de este proyecto,

se trabajó con redes VLC, de las cuales se desarrollaron dos tipos de enlaces de comunicación, al momento de implementar una plataforma con tecnología Lora/Lora WAN, se hizo posible que la conectividad de internet donde se instaló la plataforma, fuera independiente. Como conclusiones y resultados finales de la implementación, la plataforma HW/SW posee varias propiedades y características, que se encarga de proporcionar la solución a los usuarios de forma totalmente inalámbrica y, sobre todo, configurable. Por último, se ha demostrado que las tecnologías de conectividad VLC, BLE y Lora/Lora WAN pueden integrarse en una única plataforma HW/SW para aplicaciones IoT.

Según [Rose et al., 2015] el Internet de los objetos (IoT) se ha convertido en un tema importante en la tecnología, la política y la ingeniería, han causado un gran revuelo en la prensa empresarial y los medios de comunicación. La tecnología se incorpora a una amplia gama de productos, sistemas y sensores en red, aprovechando los avances en potencia informática, reducción de componentes electrónicos y redes para ofertar nuevas competencias, lo que antes era imposible. Una serie de discursos, reportes y artículos de noticias que debaten el posible impacto de la “revolución IoT”, desde nuevos modelos de negocios y oportunidades de mercado hasta preocupaciones de seguridad, privacidad e interoperabilidad técnica. Los autores dan a conocer su perspectiva sobre cómo es de suma importancia esta tecnología en grandes industrias, además nos comunica sobre las novedades que se vienen implementando.

Según [Ismael, 2018] indica que una de las redes en las que opera el Internet de los objetos es la LPWAN, un sistema inalámbrico de comunicación de banda ancha, diseñada para comunicaciones de largo alcance, buena autonomía de batería, costo de hardware baja rigidez y con un alcance generalmente mayor a 2 km, la combinación de alcance y potencia crea un sistema amigable con el medio ambiente, haciéndolo atractivo para el desarrollo de proyectos IoT. LPWAN gestiona varias tecnologías, tanto las ofrecidas en el espectro con licencia y sin licencia, como NBIoT, LTEM, LORA, SIGFOX. Dentro de estos grupos se destaca las dos últimas tecnologías dado que cada vez se encuentran en mayor apogeo, sobre todo por los beneficios que estos ofrecen por las siguientes características: la amplia duración de batería, su capacidad de transmisión y sobre todo por el bajo costo de instalación que estos representan.

De igual manera Katherine and Daniela [2019] en su trabajo “Análisis del desempeño de la tecnología LORAWAN aplicada para desarrollar el sistema de monitoreo de la calidad del aire en la Unidad Educativa de Santo Tomas Apóstol Riobamba”, destaca un sistema de iluminación automatizado se

compone tanto del hardware (parte física), como del software (parte lógica), el cual se encargará de la comunicación y gestión remota de los dispositivos, permitiendo una conexión inalámbrica entre el panel de control central y los sensores encargados de controlar la iluminación de las estancia. El control remoto de los dispositivos es generado por la interfaz gráfica, que puede programarse o activarse manualmente en tiempo real; entre sus ventajas se encuentra un entorno de fácil manejo que permite al personal familiarizarse rápidamente con el sistema. Los dispositivos ofrecidos en el sistema tienen consumo de energía reducido y mayor duración del servicio garantizado en el caso de los modelos que funcionan con pilas, lo que se traduce en un menor impacto medio ambiental y un ahorro en los costes de consumo de energía. La electricidad desempeña un papel muy importante en la vida humana, por lo que es uno de los recursos más utilizados y explotados del mundo, pero también perjudica al medio ambiente por sus efectos nocivos y el aumento de los combustibles fósiles.

Por consiguiente, [Corrente \[2020\]](#), llevó a cabo un diseño de prototipo funcional de alumbrado público, el cual se basó en tecnologías IoT. Debido a la carencia de un sistema de alumbrado que pueda contar con la capacidad de realizar un consumo eléctrico eficiente en la ciudad, de la misma forma, tampoco existía una manera de poder tomar mediciones respecto al alumbrado, ni tampoco gestionar de manera correcta el funcionamiento de las luminarias. A ello, el autor hizo hincapié en que actualmente, algunas ciudades han integrado sistemas basados en IoT, mediante los cuales se logra dinamizar el funcionamiento de las luminarias, logrando que se generen recursos eléctricos acordes y de manera correcta. Dentro de los objetivos específicos a emplear en este proyecto, se tuvo que realizar un diagnóstico sobre el sistema de alumbrado público, del cual mediante el diseño framework conceptual, sobre una propuesta de solución de alumbrado público basado en tecnologías IoT por medio de la arquitectura del sistema de alumbrado. Finalmente, el autor concluye que el sistema de alumbrado público inteligente resulta ser una opción muy potente, debido a que gracias a la robustez de su infraestructura se puede brindar varios aportes significativos.

Asimismo, otro hallazgo, en Ecuador, en la ciudad de Guayaquil, [Zavaleta \[2020\]](#), llevaron a cabo su proyecto de titulación, el cual se basa en la implementación del prototipo basado en IoT y WSN para el alumbrado público inteligente en la ciudad de Guayaquil. Mediante este proyecto los autores afirman que se logra monitorear cada una de las luminarias a través de los nodos finales, asimismo, mediante el protocolo LORAWAN, los datos que son enviados, llegan a una plataforma TTN, la cual se encargó de integrar la red al servidor IoT. Por consiguiente, a través del prototipo

realizado, mediante una regulación inteligente de las luminarias a través de dispositivos (nodos finales) conectados a la red LORA, se pueden captar datos importantes como el consumo de energía, los mensajes de encendido y apagado oportunos y la gestión de la regulación de las luminarias, pero, sobre todo, la gestión inteligente de las mismas.

Como un último hallazgo, en Puno-Perú, [Alcolea \[2019\]](#) realizó el planteamiento y la aplicación del sistema IoT para la gestión del alumbrado público en la ciudad. El sistema se conformó por un controlador que se ubicó en las unidades de alumbrado público dentro de la ciudad universitaria; a través de dicho controlador se reciben señales de los sensores, los cuales son interpretados de manera digital o analógica, y se envían datos remotamente hacia un administrador, de manera que pudiera existir un control en cuanto a los estados de las luminarias. Para finalizar, el autor concluye, que este diseño permite abrir una nueva forma de monitorear y controlar un sistema de iluminación, así mismo, afirma que, la implementación del prototipo permitió la gestión del sistema, a través de un navegador web en el cliente.

2.2. Definiciones Previas

2.2.1. Arduino

Se conoce como Arduino a una plataforma open source diseñada para la realización de alguna tarea específica a través del uso de códigos. Asimismo, se destaca que este gadget está basado en una placa de circuito impreso que contiene un microcontrolador de tipo Amtel, el cual va de modo plugin en esta placa. Las funciones del dispositivo incluyen el control de motores, sensores, dispositivos de iluminación, LEDs y servomotores para que el dispositivo pueda controlar el entorno físico.

Entre las características principales por la que se opta utilizar esta herramienta se destacan [Andrés \[2019\]](#):

- Económico: Una de las grandes ventajas que presenta este equipo es el bajo costo que este representa, el cual en comparación a placas como Raspberry pi o Thinkerboard su precio es mucho más accesible.
- El software Arduino es compatible con todos los sistemas operativos en el mercado, es decir, que se lo puede utilizar tanto en Linux, Macintosh como en Windows.
- Entorno de programación intuitivo y amigable: Otro beneficio que ofrece que esta placa es su IDE, ya que su entorno es sumamente fácil de entender e intuitivo en su manejo.
- Open Source: Tanto a nivel de hardware como en software este dispositivo es de código abierto; ya que a nivel lógico se puede manipular su estructura a libre albedrío acorde a las necesidades del usuario y en cuanto a hardware estas placas incorporan microcontroladores desmontables multicompatibles en dónde se almacena y gestiona la codificación, lo que desemboca que el Arduino sea un dispositivo de hardware extensible y de acceso libre.

Un punto a resaltar de estas placas es que existen diversos modelos y entre los que más se destacan están:

- Arduino UNO: Este es uno de los tipos más comunes y fácil de usar de las placas Arduino; utiliza el microcontrolador ATmega328P. En cuanto a sus características extrínsecas se destaca que este modelo cuenta con 14 pines de Entrada y Salida, de los cuales 6 pueden ser utilizados en la modulación por ancho de pulso (PWM) y los pines

restantes 6 son usados para los comandos analógicos y 1 para el botón de reinicio, tal se muestra a continuación en la Figura 2.1

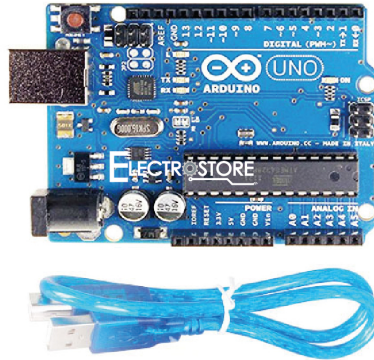


Figura 2.1: Arduino Uno. Fuente: [Store](#) [2019]

- LilyPAD Arduino: Esta placa está diseñada como tecnología textil portátil. Entre las características de mayor valía es que esta placa se la puede coser a la ropa sin dañar la placa, asimismo, es lavable. En cuanto a nivel técnico este dispositivo cuenta con placas de Entrada y Salida, sensores y pin de alimentación. Tal como se muestra en la Figura 2.2



Figura 2.2: Arduino LilyPAD. Fuente: [Lilypad](#) [2019]

- Arduino Mega: El Arduino Mega o también llamado como R3 no es más que una versión más grande que la del Arduino UNO. Cuenta con múltiples pines de Entrada y Salida y es adecuado para usarlo

como PWM. Además, cuenta con un USB, con un botón de reinicio, un conector de alimentación y otra opción de conexión esencial. Asimismo, presenta una función para conectarse con un equipo utilizando el dispositivo USB y se lo puede controlar con un adaptador de CA a CC, como se lo verá indicado en la Figura 2.3

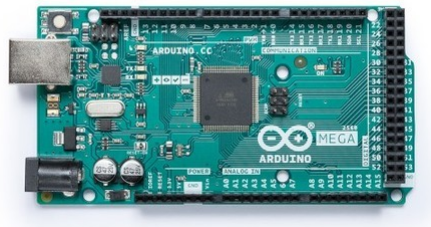


Figura 2.3: Arduino Mega. Fuente: [Store \[2019\]](#)

- Arduino Nano: En este tipo de placa Arduino se suele utilizar el procesador ATmega628 o ATmega328P y cuenta con la capacidad de conectarse a la red desde el lado del Arduino y es muy utilizada para proyectos IDE gracias a su tamaño compacto, flexible y fiable y su diseño se lo verá presentado en la Figura 2.4



Figura 2.4: Arduino Nano. Fuente: [Store \[2019\]](#)

2.2.2. Sensores

Un sensor consiste en la detección de magnitudes físicas y químicas y su conversión en señales eléctricas. En cuanto a su mecanismo se destacan los siguientes tipos de sensores [Richard Chávez Vásquez \[2019\]](#):

- Posición: Permite definir la posición del contexto en dónde se encuentra situado el sensor
- Movimiento: Distingue diversas alteraciones de movimiento en el contexto dónde se ubica el equipo.
- Magnéticos: Detecta variaciones electromagnéticas en el sector dónde se encuentra situado el sensor.
- Presión: Detecta alteraciones de presión del medio estipulado.
- Humedad: Distingue variaciones de humedad en el ambiente.
- Temperatura: Percibe alteraciones de temperatura en el entorno deseado

Sensor de corriente no invasivo SCT-013-030

La serie SCT-013 es un dispositivo que funciona como un transformador. Utiliza como fuente principal la corriente que fluye en el cable bajo prueba. Hay que señalar que la relación entre la intensidad que fluye a través del cable y la intensidad suministrada por el transductor varía entre los diferentes modelos de esta serie; este modelo en concreto es capaz de medir una corriente de aproximadamente 100A [Naylmap \[2018\]](#). Y este módulo se lo verá presentado en la Figura 2.5.



Figura 2.5: Sensor de Corriente. Fuente: [Naylmap \[2018\]](#)

Tenga en cuenta al respecto que este módulo se denomina transformador de corriente o sensor CT porque funciona como un transformador, para ello, la corriente que se mide funciona como devanado primario, y éste tiene un devanado secundario que puede llegar a tener más de 2.000 devanados, según el modelo. Una ventaja de este transductor es que al ser una pinza de corriente, no es necesario cortar o deshacer el cable para medir la corriente.

Módulo de Sensor de Luz BH1750

El BH1750 es un sensor de luz capaz de medir la intensidad luminosa en unidades LUX estándar y en cuanto a su rango de precisión este valor oscila entre 1 lx y 65535 lx [Naylmap \[2018\]](#).

En cuanto a sus características este módulo cuenta con un regulador interno de 3.3v y del mismo modo cuenta con un interfaz de comunicación de tipo I2C lo que le permite ser compatible con la mayoría de controladores y en cuanto a su conexión este módulo cuenta los pines de alimentación, pines I2C y el de dirección, tal como se detalla en la Figura 2.6.

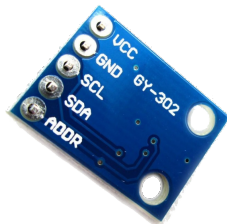


Figura 2.6: Módulo Luxómetro. Fuente: El autor

Sensor de Voltaje

Es un módulo divisor de tensión que cuenta con una resistencia oscilante de 30 K y 75 K y es capaz de medir voltajes de hasta 25 Voltios y en cuanto a su compatibilidad este sensor puede ser gestionado a través de las placas de Arduino, NodeMCU y ESP32 [Electronics \[2022\]](#). Y su diseño se verá reflejado en la Figura 2.7

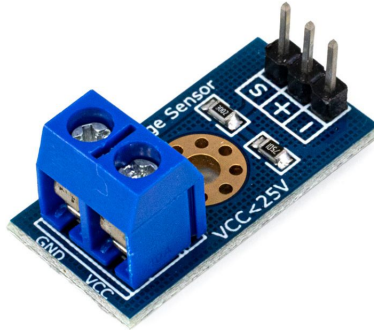


Figura 2.7: Sensor de Corriente. Fuente: [Naylmap \[2018\]](#)

Debido a su función, estos sensores se utilizan a menudo para evaluar el estado y la alimentación de la batería de los equipos en el rango de 12 a 24 V. También pueden utilizarse para medir la temperatura del agua.

Y en cuanto a sus características técnicas se tiene:

- Rango de entrada de voltaje de 0V a 25V DC.
- Rango de detección de voltaje de 24.41mV a 25V.
- Resolución analógica de tensión de 0.00489V DC.

Sensor de Movimiento DSC LC-151

El sensor LC-151 es un dispositivo diseñado para la detección de movimientos a través de un elemento pasivo infrarrojo, el cual, en conjunto con la tecnología tipo microondas es capaz de detectar movimientos en exteriores sin importar las condiciones climáticas presente, entre sus principales características están [PCEL \[2018\]](#):

- Detección con microondas fundado bajo el efecto Doppler relés normalmente abiertos y normalmente cerrados funcionan de forma simultánea.
- La altura de montaje oscila entre los 1.8 metros y los 2.4 metros y no requiere de calibración.
- Resistente a mascotas de hasta 15 kg.

- Ajuste de sensibilidad del PIR.
- Selección de intensidad de MW.
- Compensación de temperatura.
- Procesamiento de señales a través de microcontroladores.
- Interruptor de sabotaje frontal y posterior.

En cuanto a su presentación este se lo puede reflejar en la Figura 2.8.



Figura 2.8: Sensor de Movimiento. Fuente [PCEL \[2018\]](#)

En cuanto a sus especificaciones técnicas se tiene:

- Dimensiones 175mm * 70mm * 45mm.
- Peso aproximado 210gr.
- Entrada de energía de 9.6 a 16vDC.
- Frecuencia de microondas 10.525 GHz.
- Temperatura de Funcionamiento $-35^{\circ}\text{C} \pm 55^{\circ}\text{C}$.

2.2.3. Tecnología IOT

La tecnología del Internet de las cosas, denominada por sus siglas "IoT" (Internet of Things), se basa en la tecnología en la cual permite la conexión de objetos de uso cotidiano a internet en donde se intercambia,

se agrega y se procesan datos del entorno físico que los rodea, también está la capacidad de poder monitorear y reconocer eventos o cambios que ocurren ofreciendo una reacción de manera autónoma. Hoy en día el uso del IoT ha tomado mayor popularidad dentro de los avances tecnológicos y es por eso que hay que tener en cuenta ciertos requerimientos que hacen posible el uso de esta tecnología [Garcés \[2017\]](#) a continuación se detallan los conceptos básicos a considerarse:

- Los objetos con los que se trabajen pueden ser inteligentes, aunque no es obligatorio.
- Resulta conveniente que haya conectividad IP de punto a punto, a pesar de que para no todos los casos sea aplicable.
- El Cloud y el Big Data resultan indispensables cuando se use un manejo que demande mayor uso de información.
- La conectividad de la red deberá ser abierta y estándar.

Y en cuanto a su diagrama se lo puede representar como se lo muestra en la Figura 2.9

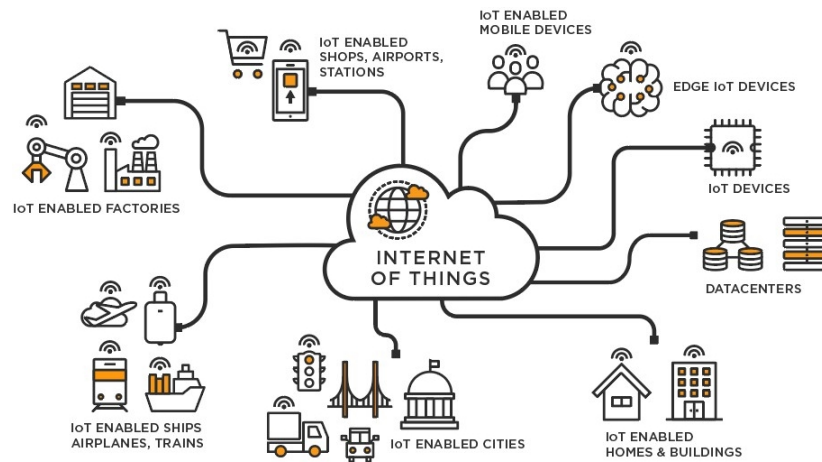


Figura 2.9: Internet de las Cosas [TIBCO \[2021\]](#)

Características de IOT

La tecnología dada por el internet de las cosas presenta modelos que ayudan en la construcción de proyectos, se forman a través de una estructura

por niveles en donde se detallan aquellos elementos que son parte de cada una de estas capas y como es su operatividad entre sí. Una de las arquitecturas que es mayormente empleada en el desarrollo de proyectos formados por IoT es el modelo conformado por 7 niveles, éste modelo es promovido por el Foro mundial de IoT [Romo-González and de los Ángeles Villalobos-Alonzo \[2019\]](#)

En la Tabla 2.1, se indica el modelo de referencia de IoT:

Tabla 2.1: Esquema general del modelo de referencia de Iot

Nivel	Nombre	Elementos
7	Colaboración y procesos	Procesos de negocio e intervención de personas
6	Aplicación	Reporte, analítica y control
5	Abstracción de Datos	Agregaciones y accesos
4	Abstracción de Datos	Almacenamiento
3	Computación en la Frontera	Elementos de análisis y transformación de datos
2	Conectividad	Comunicación y unidades de procesamiento
1	Dispositivos Físicos y Controladores	Actuadores, sensores, máquinas inteligentes, dispositivos, nodos de todo tipo

Nota. La tabla muestra el esquema del modelo de referencia IoT con sus respectivos niveles. Fuente: [Romo-González and de los Ángeles Villalobos-Alonzo \[2019\]](#)

El artificio de Internet de las Cosas está conformado por cuatro principales características y estas son [George and Velazco \[2020\]](#):

(a) Detección y actuación.

Los objetos usados cuentan con inteligencia embebida que gracias al uso de los sensores permite detectar y medir variables físicas y actuadores para así poder tomar acciones ante otros elementos que forman parte del sistema.

(b) Comunicación.

La comunicación se lleva a cabo mediante protocolos interoperables entre los objetos o aplicaciones en base a lo que el usuario requiera.

(c) Procesamiento.

Toda la información obtenida por los objetos es procesada en centros de datos locales o en nubes los cuales luego serán almacenados y analizados para proporcionar información que resulte beneficiosa para el usuario.

(d) Aplicación.

Finalmente, la información que a sido obtenida se presenta a los usuarios mediante una interfaz gráfica por medio de aplicaciones.

Componentes de un proyecto de IoT

En cuanto a los componentes de los proyectos de IoT se los presenta en la Figura 2.10.

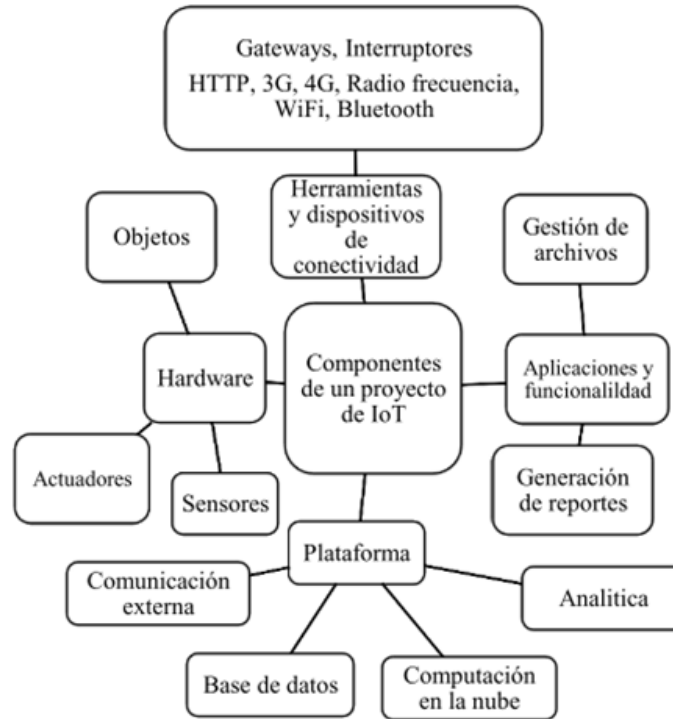


Figura 2.10: Componentes de un proyecto en IoT. Fuente: [Romo-González and de los Ángeles Villalobos-Alonzo \[2019\]](#)

2.2.4. Aplicaciones

Actualmente debido a los diversos avances tecnológicos y en la elaboración de objetos inteligentes, resulta ser productivo para la implementación de un ecosistema IoT. Existe un crecimiento en los servicios de la nube en internet y a las distintas redes de acceso lo cual hace que este tipo de tecnología abarque cada vez más, en diversos sectores en donde se desarrollan las distintas actividades cotidianas. Las aplicaciones del IoT para los diferentes sectores son [Waluyo et al. \[2021\]](#):

- Para el uso doméstico
 - Smart home.
 - Wearables
 - Coches conectados
- Para el sector industrial
 - Retail.
 - Hostelería.
 - Agricultura.
- Para el sector de la salud
 - Ciudades inteligentes
 - Gestión de suministros.
 - Monitoreo de tráfico.
 - Gestión ambiental.
 - Eficiencia energética.

Estas diversas aplicaciones apuntan a mejorar la calidad de vida, poder programar y controlar los diferentes aparatos domésticos, automatizar procesos industriales, monitorear e identificar problemas de salud, detectar desastres naturales y aquellos problemas ambientales.

2.2.5. Tecnología LORA

La tecnología LORA fue desarrollada por Semtech, LoRa (Long Range) es un tipo de plataforma inalámbrica que ofrece su trabajo a través de un largo alcance, consumo de poca potencia y mayor seguridad en el manejo de datos abarcando una mayor cobertura haciendo el uso de redes públicas y privadas (Cedillo Guzmán Marco Antonio, 2014). Dado al desarrollo de esta tecnología, hizo posible el progreso del protocolo LoRaWAN el cual es usada junto a LoRa para el manejo de datos, también puede ser reemplazada por cualquier protocolo que cumpla con las especificaciones dadas por el desarrollador. LoRa es la capa física empleada para crear conexiones para la comunicación a larga distancia que son de acceso abierto, permitiendo que cualquier entidad participe en el desarrollo de esta tecnología, mejorando la infraestructura de red y de servicios [Carolina and Ortiz \[2020\]](#).

Utiliza un patrón de modulación orientada al IoT; hace uso de banda ISM aunque tiene la capacidad de operar dentro de cualquier frecuencia por debajo de 1GHz [Hernández \[2020\]](#).

La tecnología LoRa está formada de la siguiente manera:

- Dispositivo final.

Es aquel que conecta todos los sensores a la red los cuales recolectan información en base al sistema empleado.

- Gateway.

Son aquellas estaciones en las cuales se transmiten los datos a la red de internet, éstos pueden recoger información proveniente de distintos dispositivos finales.

- Servidor.

El servidor se encarga del procesamiento de datos provenientes del Gateway.

Características Las características principales son [Hernández \[2020\]](#):

- Conexiones a costos bajos, bajo consumo de potencia, robustez ante las interferencias y resistente al efecto Doppler, todo esto gracias a que su esquema es una variación de la modulación DSSS o también denominado como Chirp Spread Spectrum (CSS).
- Opera en bandas con frecuencias de 915MHz para EE.UU., 868MHz en Europa y 433Mhz para Asia.

- La señal que envía consume un mayor ancho de banda del necesario, pero permite recibir varias señales simultáneamente a diferentes velocidades.
- Comunicación eficiente usando la mayor velocidad posible gracias a su método de ajuste dinámico (ADR).
- Tiene velocidades que van desde 0,3kbps para SF12 hasta 5486kbps para SF7.
- Emplea 6 factores de ensanchamiento y cada uno de éstos se los define en base a la relación entre la potencia y la tasa de transferencia.
- La sensibilidad de recepción que maneja es de -137dBm para SF12 y -123dBm para SF7.
- El alcance que tiene para SF12 es de 14km y para SF7 es de 2km.

En la Figura 2.11 se presenta un esquema de los elementos involucrados en una red LORA.



Figura 2.11: Componentes de un proyecto en IoT. Fuente: [ALFAIOT \[2019\]](#)

A continuación, en la Tabla 2.2 se presentan las distintas frecuencias que se emplea en función a la región:

Tabla 2.2: Frecuencias por regiones

	Europa	Norteamérica	China	Corea	Japón	India
Frequency band	867 - 869 MHz	902 - 928 MHz	470 - 510 MHz	920 - 925 MHz	920 - 925 MHz	865 - 867 MHz
Channels	10	64 ÷ 8 ÷ 8	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico
Channel BW Up	125 / 250 KHz	125 / 500 KHz	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico
Channel BW Dn	125 KHz	500 KHz	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico
TX Power Up	+14 dBm	+20 dBm typ (+30 dBm allowed)	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico
TX Power Dn	+14 dBm	+27 dBm	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico
SF Up	7-dic	7-oct	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico
Data rate	250 bps - 50 kbps	980 bps - 21.9 kbps	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico
Link Budget Up	155 dB	154 dB	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico
Link Budget Dn	155 dB	157 dB	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico	En definición por Comité Técnico

Nota. Frecuencias por regiones. Fuente: [Carolina and Ortiz \[2020\]](#)

Arquitectura de Red LORA

En la Figura 2.12 la arquitectura de la red LORA, utiliza la topología de una red en estrella de un solo salto, dicha topología permite una sencilla implementación y gestión. Está formado por gateways que retransmiten la información recibida por medio de los dispositivos finales y van hacia un servidor de red [Caballero \[2020\]](#). LoRa tiene cuatro partes en la red:

- Los nodos finales recopilan la información de los sensores, transmiten los datos upstream y downstream y reciben comunicaciones del servidor de aplicaciones. Los terminales utilizan comunicación inalámbrica de un solo paso con uno o más puertos.
- El Concentrador /Gateway actúa como un puente transparente y reenvía datos bidireccionalmente entre los nodos finales y el servidor upstream.
- El servidor web se enlaza a varios puertos mediante una conexión TCP/IP confiable, ya sea por cable o inalámbrica; eliminando mensajes duplicados; decidir qué puerto responder a los mensajes de punto final; y administre la tasa de transferencia de datos del nodo final, la tasa de datos adaptativa (ADR), para maximizar la capacidad de la red y prolongar la duración de la batería del nodo final.
- El servidor de la aplicación recopila y analiza los datos de los nodos finales y determina las acciones del nodo final.

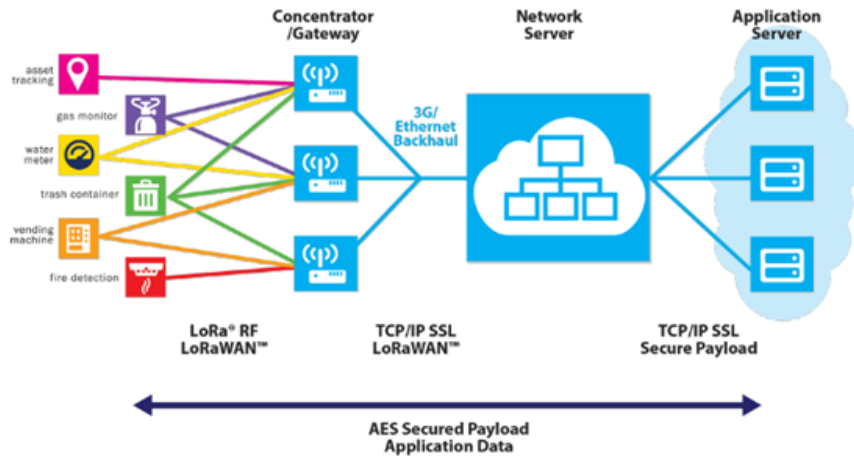


Figura 2.12: Arquitectura de red LoRa. Fuente: [Cárdenas et al. \[2018\]](#)

2.2.6. LORAWAN

LoraWAN es el estándar de LPWAN, es una red aérea extensa de muy bajo consumo. Además, LoraWAN describe la arquitectura y el protocolo de comunicación dentro del sistema utilizado en la red. Los protocolos y la arquitectura son importantes para determinar la duración de la batería de los nodos, la calidad del servicio que prestan, la capacidad de la red así como la seguridad y la diversidad de las aplicaciones que forman parte de ella [Viñas \[2018\]](#).

Según [Caballero \[2020\]](#) Lorawan, se trata la alianza LoRa ha desarrollado abiertamente un protocolo de red para comunicarse con los dispositivos LoRa y gestionarlos mediante la tecnología LoRa, es decir, la capa de medios. Se ha establecido que LoRaWAN es una especificación para LPWAN (Low Power Wide Area Network), que corresponde a la capa 2 (conexión de red). Esto se llama Control de Acceso al Medio (MAC). Se puede decir que LoRaWAN se encarga de gestionar y combinar los canales y los parámetros de conexión (canal, ancho de banda, cifrado de datos, etc.) de los distintos dispositivos LoRa. LoraWAN es un protocolo diseñado específicamente para la tecnología LoRa, emplea frecuencias de radio no autorizadas dentro de las bandas industriales, científicas y médicas (ISM) para permitir una comunicación de bajo consumo y largo alcance entre sensores remotos y puntos de acceso en red. Este enfoque estandarizado para construir una red de área amplia de baja potencia

(LPWAN) permite configurar rápidamente redes de IoT públicas y privadas en cualquier lugar, utilizando hardware y software bidireccional seguro, interoperable, móvil y consciente de la ubicación. La especificación puede descargarse gratuitamente del sitio web de la Alianza LoRa. Utiliza el espectro ensanchado denominada chirp, el cual es una modulación de espectro ensanchado en la que se genera una señal chirp que cambia de frecuencia entre f_0 y f_1 (con $f_0 < f_1$) en un intervalo de tiempo T , y todos los chirps generados se producen en el mismo intervalo de tiempo. Hay dos tipos de chirp: el ascendente, en el que la frecuencia aumenta de un mínimo a un máximo (f_0 a f_1), y el descendente, en el que la frecuencia disminuye de un máximo a un mínimo (f_1 a f_0). Estos chirridos se dividen en chips para codificar la información. Esta modulación aumenta la robustez frente a las interferencias y los efectos de desvanecimiento Dopplery. El factor de dispersión (SF), que en el caso de LoRa puede tomar valores discretos entre 7 y 12, determina el número de chips utilizados para representar un símbolo (2 SF) y el número de bits que se utilizarán (SF). Un valor alto de SF puede aumentar la SNR, la sensibilidad y el alcance de la transmisión, pero también aumenta el tiempo de emisión (ToA) de la misma. El paquete de transmisión en LoRa contiene de nuevo una tasa de código de $4/5$ a $4/8$, para manejar la tasa de error del paquete, con la siguiente ecuación que puede utilizarse para calcular la tasa de datos:

$$R_b = SF * BW / 2^{SF} * CR$$

SF= Spread Factor.

BW= ancho de banda utilizado.

CR= tasa de código utilizado.

Clases LoRaWAN

En las redes LoRaWAN, los dispositivos se dividen en clases según las funciones que soportan. Estas clases son A, B y C. A es la más pequeña y puede ampliarse para incluir a B y C. Las tres categorías pueden comunicarse en ambas direcciones e iniciar la descarga de datos al servidor a través de la pasarela. Se diferencian en el momento en que aceptan la información recibida del servidor. Como son:

Clase A: estos dispositivos ofrecen el mayor ahorro de energía ya que solo entran en modo escucha (ventana RX) después de la transmisión de un dato correctamente al gateway.

Clase B: estos dispositivos eliminan la restricción de recepción de datos con la necesidad de envío de un paquete. Tienen la ventana de recepción con base a tiempos predeterminados por el gateway. Mediante el envío de beacons por parte del gateway, este se sincroniza con el dispositivo final para planificar los tiempos de las ventanas de recepción. Al estar escuchando más tiempo presentan un mayor consumo de energía que los dispositivos de clase A.

Clase C: Estos dispositivos ofrecen el menor ahorro de energía debido a que siempre están en modo escucha y solo se interrumpe cuando se produce una transmisión.

Estructura de LoRaWan

Tal como se lo muestra en la Figura 2.13, los nodos finales se comunican con diversos puntos de accesos a través de Lora. Estos Gateways tienen la misión de reenviar la información de un servidor red a través de diversos modos de conectividad ya sea de manera inalámbrica o por ethernet. De este modo estos AP se convierten en redirecciones convirtiendo los servidores de red como los responsables de decodificando y generando los paquetes que los dispositivos finales recibirán CÁRDENAS [2018].

Esta arquitectura LoRaWAN se basa en una topología de red en forma de estrella, que se recomienda para preservar la vida de la batería de los dispositivos cuando se requiere una conexión de gran alcance.

En la Figura 2.13 se presenta la topología de estrella LoraWAN.

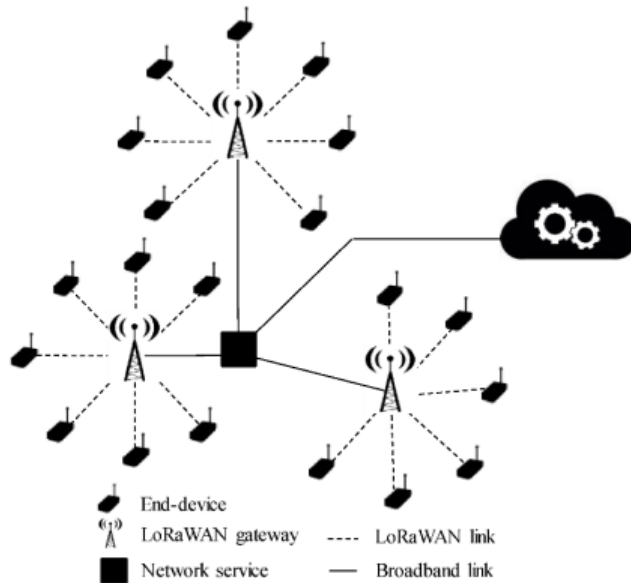


Figura 2.13: Topología de estrella LoRaWAN. Fuente: CÁRDENAS [2018].

El protocolo LoRaWAN tiene tres clases de nodos. Los nodos finales varían en función de la aplicación y los requisitos del servicio. La elección de un tipo u otro depende de los requisitos de ahorro de batería y latencia del enlace descendente. El primer tipo son los denominados nodos de clase A, que son los más utilizados y se caracterizan por no pasar al modo de escucha hasta que se han enviado los datos a la pasarela y son los más eficientes energéticamente. El segundo tipo son los nodos de clase B. En este tipo de dispositivos, las ventanas de recepción se basan en horarios pre-establecidos con la pasarela, que puede utilizar una batería o una fuente externa, en función de los horarios de recepción asignados. Por último, están los nodos de clase C. Estos nodos están siempre en modo de recepción y no tienen latencia, pero son los que más energía consumen Viñas [2018].

Topología de LoRaWan

Dentro de los sistemas de LoRa se pueden encontrar dos tipos de topologías diferentes de red, los cuales variarán de acuerdo al objetivo para el que se diseñó esta red, y estas son:

- Topología de tipo punto a punto: este diseño es ideal cuando no sea necesario realizar conexiones con redes TCP/IP y asimismo, que no se

tenga que emplear muchos nodos en dicha red, tal como se lo puede apreciar a continuación [Rafael and Jiménez \[2019\]](#):

En la Figura 2.14 se presenta la Topología punto a punto en un sistema de LORA.



Figura 2.14: Topología punto a punto. Fuente: [Rafael and Jiménez \[2019\]](#).

- Topología de tipo estrella: en cuanto al uso de este diseño se lo opta cuando se necesiten que los nodos se conecten a uno o más gateways a través de un enlace LoRaWAN y estos deban conectarse al servidor principal mediante una comunicación TCP/IP. En cuanto a su diseño se lo puede representar de la siguiente manera.

En la Figura 2.15 se presenta la Topología punto a punto mediante un servidor.

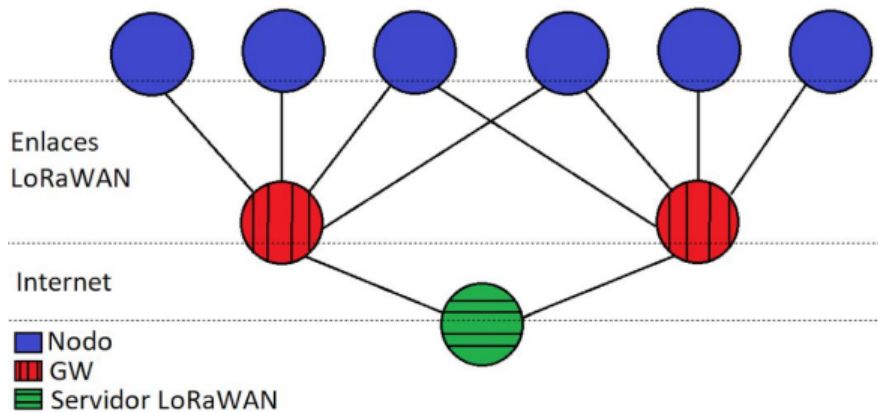


Figura 2.15: Topología punto a punto. Fuente: [Rafael and Jiménez \[2019\]](#).

Seguridad de LoRaWan

La conexión a LoRaWAN requiere que los nodos utilicen una serie de tokens y números de identificación para garantizar el correcto funcionamiento de la red y la integridad de la seguridad. Los métodos participativos incluyen el uso del protocolo de activación por aire LoRaWan (OTAA) y el protocolo de activación personalizada LoRaWan (ABP). Los parámetros necesitados por OTAA son: identificador de fabrica (DevEUI), que hace a cada dispositivo único; identificador de aplicación único (AppEUI), la dirección de 64 bits se utiliza para clasificar los dispositivos por aplicación; clave secreta (AppKEY) AES de 128 bits compartida entre el dispositivo y la red, los pasos para llevar a cabo este proceso se detallarán a continuación [CÁRDENAS \[2018\]](#):

1. El nodo solicita una conexión de red con datos de configuración y abre una ventana de descarga.
2. El Gateway recibe la solicitud y se la envía al servidor.
3. El servidor comprueba que el nodo este dado de alta y la llave de encriptación sea la correcta.
4. Si los datos son correctos, se crea una sesión temporal que gateway envía al nodo.
5. El nodo recibe la sesión temporal y puede enviar datos a la red.

2.2.7. LPWAN

Las WAN de bajo consumo, también conocidas como redes de bajo consumo", son tecnologías que tienen el potencial de reducir el consumo de potencia y energía de forma relativamente importante en comparación con otras redes de larga distancia. Estas redes comenzaron con el Internet de las cosas (IoT), que creó la necesidad de conectarse y permanecer conectado a través de Internet. Las LPWAN pueden enviar y recibir pequeñas tramas de datos a intervalos irregulares, lo que permite a sensores como los dispositivos electrónicos transmitir datos a distancias de varios kilómetros [Rosa \[2018\]](#).

Entre sus características se detacan [Almeida et al. \[2020\]](#):

- Baja potencia: los dispositivos que se encuentran conectados a la red.
- Área amplia: gracias al bajo consumo, la distancia de las redes inalámbricas es mayor, gracias a rango de operación de cobertura, llegando inclusive a superar los 2km en las áreas urbanas.

- Transmisiones densas: estas pueden ser usadas en ciudades grandes o grandes edificaciones como apoyo a la iluminación inteligente, redes inteligentes y seguimiento de los activos.
- Monitoreo (largo plazo): el monitoreo de los dispositivos y los sensores, los cuales pueden ser monitoreados por varios años, no obstante, esto dependerá del mantenimiento y cuidado, estos pueden durar más o menos que el promedio

2.2.8. Plataforma TagoIO

Es una plataforma web que trabaja el cien por ciento en la nube y ofrece un alto nivel de trabajo en el monitoreo de distintos entornos, a través de dispositivos IoT. A través de esta plataforma el usuario tiene la ventaja de poder observar aquellos datos que fueron recolectados mediante un panel que permite construir sus aplicaciones [Lucas \[2019\]](#). TagoIO ayuda en la integración de la administración de dispositivos, almacenamiento, visualización, análisis y demás situaciones que permiten el monitoreo de dicho entorno en el cual se trabaja. Permite una rápida construcción de una solución IoT; dichas soluciones pueden estar orientadas al campo de la automatización industrial, telemática, irrigación inteligente, entre otros. Asimismo, se detalla que para la creación de soluciones IoT se tiene que seguir las siguientes etapas:

1. Conexión del dispositivo mediante Wi-Fi.
2. Lectura y envío de datos a la plataforma.
3. Inicio de la construcción de la aplicación e integrar las soluciones mediante los datos adquiridos.
4. Creación y administración de usuarios y dispositivos. Tal como se lo muestra en la [Figura 2.16](#)

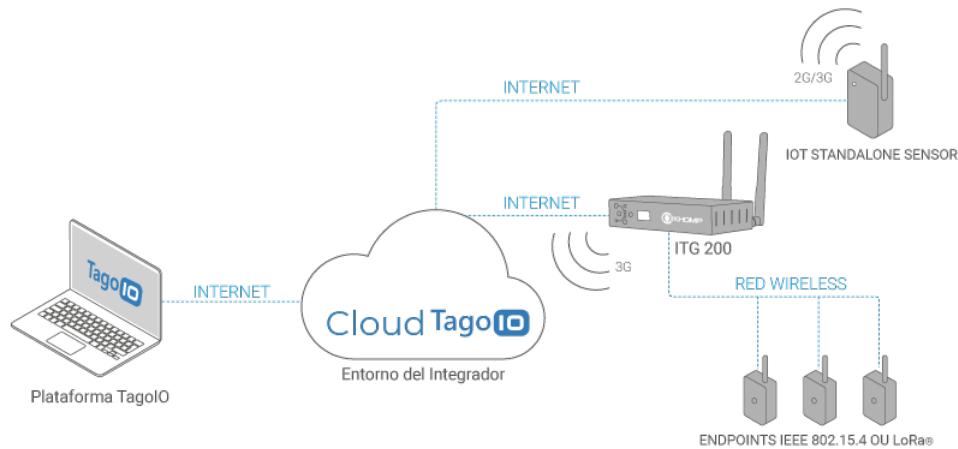


Figura 2.16: Nube TagoIO: Modelo de Aplicación. Fuente [Lucas \[2019\]](#)

En cuanto a sus funciones principales están:

- . Configuración de los dispositivos IoT.
- . Configuración de un banco de datos.
- Análisis y automatización de flujos.
- Sincronización de eventos y ejecución de acciones.

Y en ventaja se destaca los siguientes puntos:

- Capacidad para la conexión a cualquier dispositivo.
- Uso y creación de plantillas o tableros de acuerdo con las necesidades.
- Fácil administración de datos y archivos.
- Permite el análisis de datos y la automatización de flujo.
- Mediante mensajes o notificaciones, permite conocer los eventos presentados y tomar acciones ante los mismos.
- Creación de aplicaciones que ayudan al fácil manejo del usuario.

2.2.9. Controlador Fuzzy

El controlador Fuzzy se trata de una forma de agrupación de los sistemas expertos, es muy popular ya que se basa en la dificultad de moldear muchos sistemas por medio de ecuaciones matemáticas. Por otro lado, este tipo de controlador, es sin duda muy útil ya que radica en aprovechar al máximo el conocimiento de un experto, que este exento de requerir a modelar matemáticamente un proceso. Los controladores difusos son una extensión de la lógica clásica y están diseñados para permitir el razonamiento de conceptos vagos. Las aplicaciones incluyen el desarrollo de sistemas expertos difusos para el reconocimiento y la clasificación de patrones, la agrupación difusa, las pruebas difusas, la supervisión, el diagnóstico y la optimización [Chen et al. \[2020\]](#). En cuanto a su estructura se lo puede reflejar en la Figura 2.17

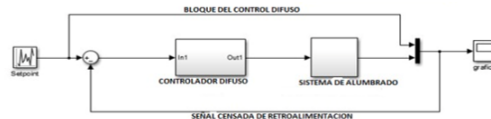


Figura 2.17: Modelo de Fuzzy. Fuente: El autor

Necesidades del controlador Fuzzy

Dentro de un controlador Fuzzy se usa la identificación de patrones y el análisis de las decisiones en comparación a otros métodos tradicionales. Su conceptualización de control es única y tiene una mayor facilidad al momento de lograrla con un propósito preciso sin tener en cuenta un modelo matemático. Sin embargo, en diferentes entornos, existen diferentes estándares de iluminación; por lo tanto, un controlador difuso puede realizar un control preciso de la iluminación solo a través del conocimiento experto y la ley empírica. The Fuzzy Design of the Energy-Saving Lamps base on the Internet of Things

Lógica Fuzzy

La lógica difusa se utiliza para predecir la duración de la disponibilidad de la batería en cada lámpara. La variable de entrada en fuzzy es la energía restante cuando se toma, la cantidad de energía necesaria para encender las luces por minuto y la cantidad de capacidad de la batería para recibir energía de la celda solar. El primer parámetro en el método difuso es la condición de las reservas de energía restantes en la batería. La condición se divide en

tres conjuntos que son bajo, suficiente y completo. La condición baja está representada por la variable low_c que tiene una membresía de 0-40 % de potencia, luego un promedio de 40-80 % de potencia está simbolizado por la variable $avgc$, y alta que está entre 80-100 % de potencia con el símbolo igc . El primer parámetro de entrada se puede presenciar en la Ecuación:

$$low_c : 0 \leq / \leq 40$$

$$\mu_x = avgc : 41 \leq / \leq 80$$

$$high_c : 81 \leq / \leq 100$$

El segundo parámetro de entrada es la cantidad de intensidad de luz y energía térmica recibida por el panel solar por la lámpara, esto afecta la cantidad de voltaje y energía cargada a la batería. Si la cantidad de intensidad de luz y energía térmica absorbida es pequeña, se puede concluir que la batería no está completamente cargada. Por el contrario, si recibe demasiada luz solar, la batería también puede experimentar un exceso de capacidad. La ecuación dos es una ecuación para analizar la cantidad de voltaje producido por la energía solar térmica (μ_y). El conjunto de parámetros de entrada se puede dividir en tres, a saber, d para intensidades de calor por debajo del estándar que debe aceptarse, f para la intensidad de luz de acuerdo con los estándares que deben aceptarse y S para la intensidad de luz recibida supera los estándares que deben aceptarse. Ser aceptado, esta formulación se la puede presenciar en la Ecuación:

$$dL_r < S_d$$

$$\mu_y = f : 41 \leq / \leq 80$$

$$s : 81 \leq / \leq 100$$

Capítulo 3

Marco Metodológico

La base del proyecto consiste en analizar las señales de control recibidas del lugar de pruebas, la información procedente de la base de datos adquiridas por los sensores para evaluar la funcionalidad y la viabilidad enviados al sitio web. Para monitorear y almacenar los datos del sistema de iluminación, se diseña una interfaz web usando tecnología IoT, el cual permite detectar fallas. En este capítulo se explica detalladamente el proceso de comunicación del Gateway Dragino con el servidor TTN. Se diseña por medio de TagoIO cuatro estaciones de medición remota llamada Lampara1, Lámpara 2, lampara 3, lampara 4 los cuales se mide la corriente y luminosidad de una lámpara dimerizable junto con la programación del software Arduino, se implementó un sistema de control fuzzy para controlar la cantidad de luxes enviadas a las diferentes áreas de un parque.

3.1. Diseño del experimento

El diseño del presente trabajo de investigación ha optado por utilizar tecnología Open Source, en dónde se destacan fabricantes como Arduino y Dragino y esto se debe a las prestaciones tecnológicas, interoperabilidad, disponibilidad, costos y sobre todo por la curva de aprendizaje que requiere utilizar estas herramientas.

Del mismo modo se destaca los equipos utilizados para la recolección y envío de datos. Y, para esta propuesta se optaron por equipos como el sensor de corriente SCT-013-013; el módulo luxómetro BH1750; el sensor de voltaje; el receptor JFL; el sensor de movimiento y como placa de gestión se optó por el Arduino Nano y finalmente se diseñó un dashboard con TAGO IO e Inkscape para visualizar la data conseguida. Paradigmas y herramientas que se ampliarán en los ítems siguientes:

3.1.1. Diseño de la Red LoRa

Para el diseño de la Red LoRa se optó como diseño de red la topología estrella, puesto que para la elaboración de la propuesta se requiere llevar a cabo el paradigma de modelo cliente servidor, el cual permitirá obtener ciertos índices del sistema de luminarias led, mediante el uso de sensores para así posteriormente gestionar estos valores a través de un dashboard. Y, para cumplir este objetivo se requiere cumplir el siguiente planteamiento:

- Extraer las variables necesarias por medio de los sensores que se encuentran situados en cada una de las luminarias. Los cuales formarán una red Lora tipo malla.
- Posteriormente se deberá llevar a cabo la etapa de recopilación y transferencias de datos de los equipos de cada nodo instalado bajo la tecnología LoRa.
- Ya recopilada esta información se deberá llevar a cabo la comunicación del Gateway, el cual por medio de la conexión Ethernet se instalará un HMI y a priori se enviará esta data a la nube IoT.
- Y finalmente toda esta información se verá reflejada por medio de un sistema web en tiempo real.

En síntesis, se plantea utilizar como arquitectura el diagrama presentado en la Figura 3.1, ya que el proyecto al tratarse de automatización en un sistema de alumbrado en un parque estará compuesto de luminarias led

dimerizables 0-10V las cuales se obtendrá las variables de luxes voltaje y corriente de cada una de ellas, además su controlador estará en cada luminaria. Al contar con una estructura IoT, se deberá disponer de un módulo o nodo de comunicación IoT, en cada una de las lámparas, el cual enviará al Gateway o puerta de enlace la data adquirida por los sensores conectados en cada una de las luminarias. Las variables de cada lámpara a monitorear se enviarán a la nube IoT mediante el paradigma del Internet de las Cosas se podrá tanto monitorizar en tiempo real el estado de estas variables, mediante el Dashboard creado para la visualización del circuito de alumbrado a monitorear, donde nos refleja alarmas y además nos permite tener un historial del estado de las lámparas. Nos da la facilidad de conectarnos desde cualquier dispositivo electrónico ya sea pc, laptop celular o Tablet, mediante un link desde cualquier momento y lugar. Además, se puede visualizar mediante un display las variables de cada lámpara en el mismo sitio así dando la facilidad al técnico para ver el estado en que se encuentra la luminaria.

Se visualizará mediante el HMI el estado de funcionamiento del control Fuzzy en manual o auto, y el funcionamiento de los sensores de movimiento.

De acuerdo con el contexto presentado, en este caso regular las condiciones de iluminación, voltaje y corriente ante cualquier eventualidad intrínseca o extrínseca que pueda llegar afectar los sistemas de alumbrado público en parques y jardines.



Figura 3.1: Diagrama de la red LoRa. Fuente: El autor

Nodos Transceptores

Sensor de Corriente no invasivo SCT-013-030. -

Este equipo (Figura 3.2) permite realiza medidas dentro de un rango de 3A. Este modelo en específico tiene una resistencia de carga interna, el cual nos entrega una salida de voltaje.



Figura 3.2: Sensor de Corriente. Fuente: El autor

En cuanto a su conexión con el Arduino o con cualquier otra placa PCB, se requiere seguir la siguiente conexión que se verá reflejada en la Figura 3.3.

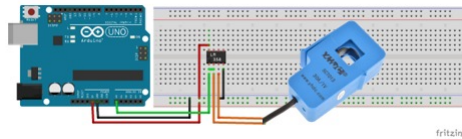


Figura 3.3: Conexión del sensor de corriente. Fuente: El autor

Mientras que para su codificación se requiere realizar lecturas de la entrada analógica.

```
int Sensordecorriente analogRead(A0); //pin Analógico 0
```

Posteriormente, se requiere calcular el voltaje del sensor y esto se lo realiza a través, del siguiente código.

```
float Sensordecorriente = analogRead (A0) * (1.1 / 1023.0);
```

A continuación se calcula corriente, el cual corresponde al producto del voltaje del sensor multiplicado por 30A dividido para 1V.

```
float corriente=SensordeCorriente*30.0;
```

Y finalmente se mostrará el resultado en la pantalla serial mediante la siguiente línea de código, tal como se lo muestra en la Figura 3.4

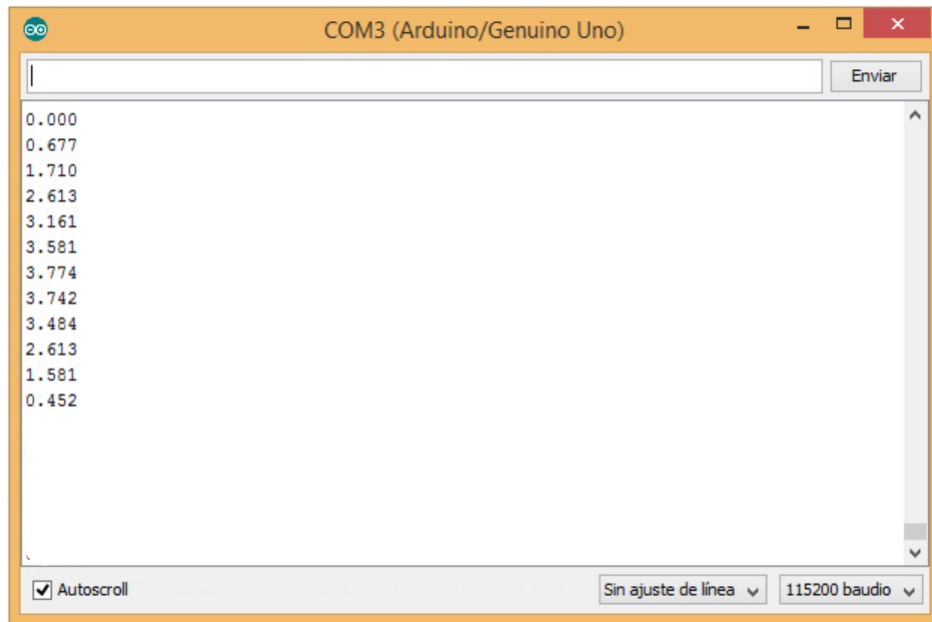


Figura 3.4: Resultado de la medición de corriente. Fuente: [Naylmap \[2018\]](#)

Módulo Luxómetro BH1750. -

EL módulo BH1750 (Figura 3.5), es un sensor de luz capaz de entregar valores de medición en Luxex (lumen / m^2), el cual es una unidad de medida estándar utilizada para cuantificar el nivel de iluminación (iluminancia). Tiene alta precisión y un rango ente 1 – 65535 lx el cual es configurable.



Figura 3.5: Módulo Luxómetro. Fuente: El autor

Para su conexión con el Arduino se debe realizar el diagrama reflejado en la Figura 3.6

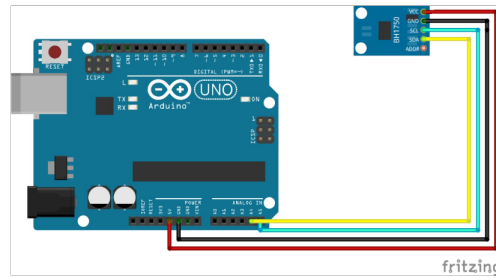


Figura 3.6: Módulo Luxómetro. Fuente: El autor

Pero se destaca que su conexión varirará de acuerdo al controlador que se esté empleando y para esto se tendrá que guiar con la Tabla 3.1:

Tabla 3.1: Esquema de conexión entre controladores y el Módulo BH1750

Puertos	Arduino Uno, Nano, Mini	Arduino Mega, Due	Arduino Leonardo
VCC	5V	5V	5V
GND	GND	GND	GND
SCL	A5	21	3
SDA	A4	20	2
ADDR	No conectado	No conectado	No conectado

Nota. La tabla muestra el esquema de conexión que se deberá seguir entre el Módulo BH1750 en diferentes controladores. Fuente: [Naylmap \[2018\]](#)

En cuanto a su codificación este sensor requiere que se exporte una librería de terceros en este caso es la **BH1750.h**, la cual podrá ser descargada desde <https://github.com/claws/BH1750> y esta librería permitirá inicializar la comunicación I2C del Arduino y configurará la resolución del sensor BHT175.

Para su funcionamiento se deberán seguir los siguientes pasos:

1. Invocar la librería BH1750.h

```
#include<BH1750.h>
```

2. Crear una instancia a la librería

```
BH1750 Luxometro
```

3. Inicializar la lectura del sensor

```
Luxometro.begin(BH1750_CONTINUOUS_HIGH_RES_MODE);
```

4. Realizar la lectura del sensor

```
uint16_t lux = Luxometro.readLightLevel();
```

5. Presentar los resultados a través del monitor serial

```
Serial.print(lux);
```

Sensor de Voltaje. -

El sensor de voltaje (Figura 3.7) es un dispositivo que sirve para calcular y controlar la tensión de un objeto. El sensor de tensión es capaz de determinar los niveles de tensión alterna y continua. La entrada de este sensor puede ser una tensión, mientras que la salida puede ser un interruptor, una señal de tensión analógica, una señal de corriente, una señal de audio, etc. Un sensor es esencialmente un dispositivo capaz de detectar o identificar y responder a ciertos tipos de señales eléctricas u ópticas. El uso de la detección de tensión y corriente es una excelente alternativa a los métodos tradicionales de medición de tensión y corriente.

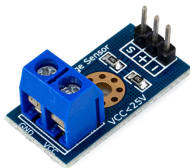


Figura 3.7: Sensor de Voltaje. Fuente: El autor

Para su conexión se sugiere seguir el diagrama de conexión presentado en la Figura 3.8:

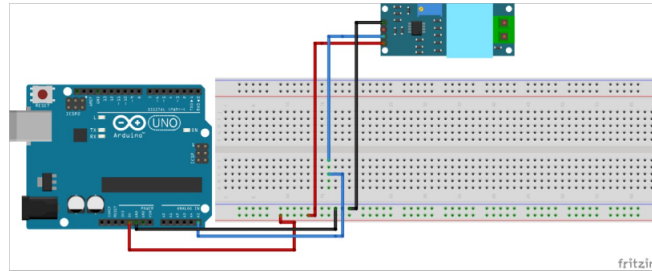


Figura 3.8: Conexión de Voltaje. Fuente: El autor

En cuanto a su codificación se debe realizar el siguiente algoritmo.

1. Inicializar la variable que identifique al sensor en el puerto A0.
`int Sensor = A0;`
2. Realizar el cálculo respectivo para la determinación del voltaje.
`float voltaje = (float)25*analogRead(A0)/1023;`
3. Presentar el cálculo de voltaje en el monitor Serial.
`Serial.println(voltaje);`

Sensor de movimiento. -

Un sensor de movimiento (Figura 3.9) es un equipo electrónico diseñado para activar algún sistema o mecanismo cuando detecta movimiento en el área o entorno en el que está instalado. Suelen utilizarse para optimizar el consumo de energía y la eficiencia de diversos sistemas, como la ventilación, la iluminación y el aire acondicionado en hogares y oficinas, pero también en el sector de la seguridad.



Figura 3.9: Sensor de Movimiento. Fuente: [Systems \[2022\]](#)

Receptora JFL. -

El siguiente equipo es una receptora de 2 canales que trabaja a 433 Mhz con un alcance promedio de 100mts en línea de vista. Se va a utilizar este módulo inalámbrico para la activación del modo dimerización en la tarjeta controladora de cada lámpara. En cuanto a su forma se lo puede presenciar en la Figura [3.10](#)



Figura 3.10: Receptor JFL. Fuente: El autor

3.1.2. Diseño del Dashboard

Como prefacio se indica que el Dashboard es un interfaz de usuario que permite presentar los datos capturados en tiempo real desde un servidor [Ferrandez-Pastor et al. \[2021\]](#), y en este caso se optó por la plataforma TagoIO.

TagoIO es una plataforma que permite construir y desarrollar soluciones IoT de forma ágil y flexible lo que permite inculcar en áreas como automatización industrial, irrigación inteligente, localización interna de depósitos, composición, refrigeración y telemática [Ferrandez-Pastor et al. \[2021\]](#) y en este caso permitirá mostrar los datos capturados desde cualquier dispositivo que tenga acceso a Internet.

Para esto es necesario crear un usuario en la página de Tago y una vez realizado este proceso se deberá ingresar a la ruta ->Applications ->Integrations ->Webhooks y posteriormente seleccionar la opción de Tago IO, tal como se lo indica en la Figura [3.11](#)

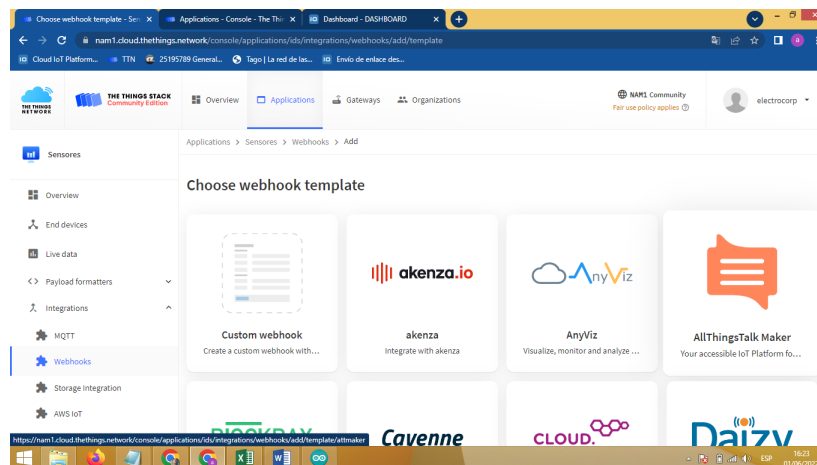


Figura 3.11: Panel de selección para la vinculación del TTN con Tago. Fuente: El autor

En Webhook ID se deberá agregar un nombre y en Authorization se deberá agregar un token que se genera al momento de crear una cuenta en Tago IO, como se lo indica en la Figura [3.12](#)

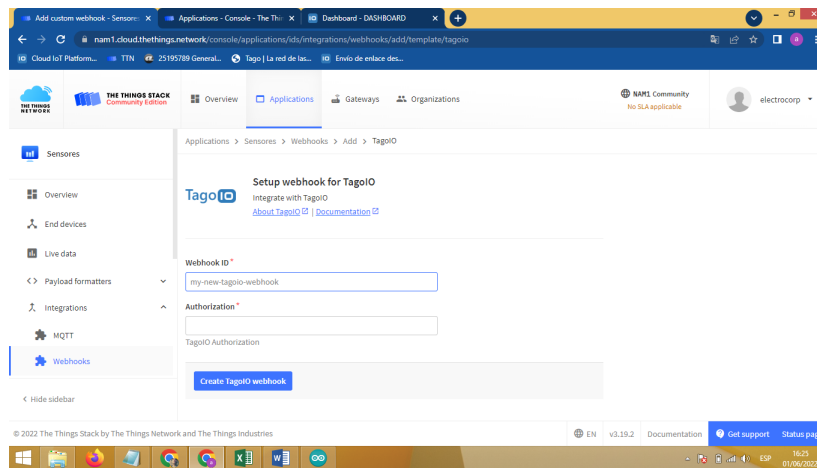


Figura 3.12: Creación del proyecto en Tago IO. Fuente: El autor

Tago IO

El primer paso para trabajar en la plataforma Tago Io es crear un ID, por lo que es necesario dirigirse a Devices -> Authorization, tal como lo muestra en la Figura 3.13

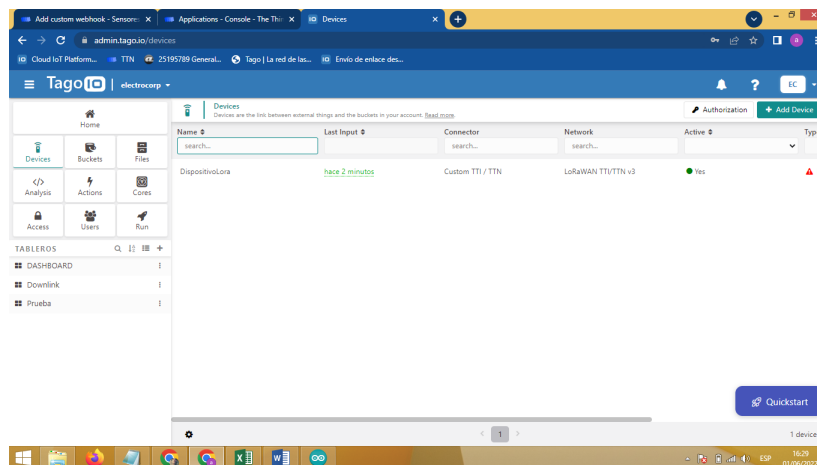


Figura 3.13: Creación del token de Tago IO. Fuente: El autor

Una vez agregado el nombre se debe dar clic en Generate para así generarlo de manera automática, cabe resaltar que este id permitirá relacionar este sistema al TTN, tal como lo indica la Figura 3.14

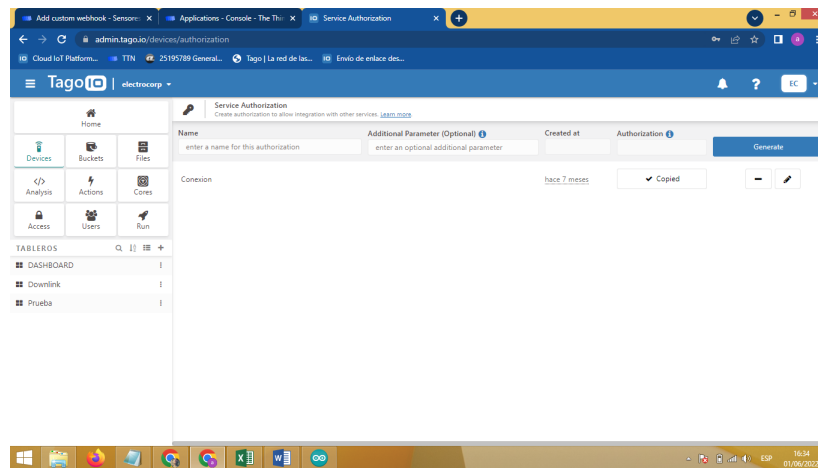


Figura 3.14: Generación del ID para vinculación al TTN. Fuente: El autor

Una vez inicializado el Tago IO, se debe agregar un dispositivo y para esto se debe dirigir a la siguiente ruta Device -> LoraWan TTI/TTN v3 -> Custome TTI/TTN, del mismo modo que se lo refleja en la Figura 3.15

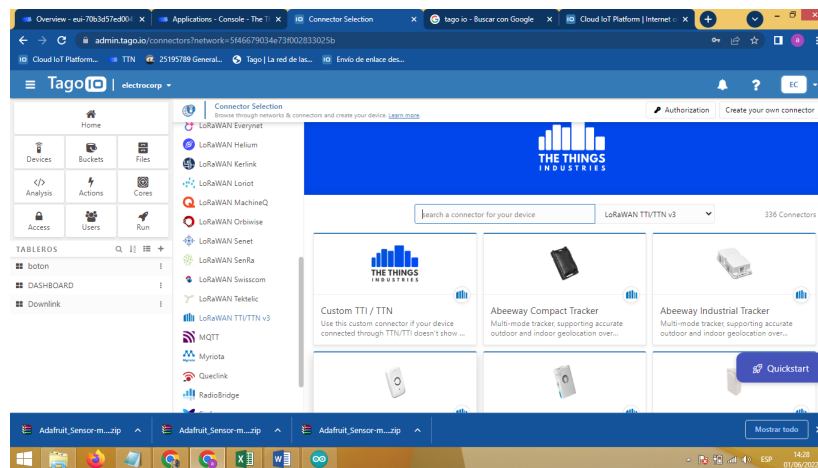


Figura 3.15: Incorporación del dispositivo a la plataforma. Fuente: El autor

Ya seleccionado el dispositivo se debe agregar un nombre y el id único del Dragino, tal como se lo muestra en la Figura 3.16.

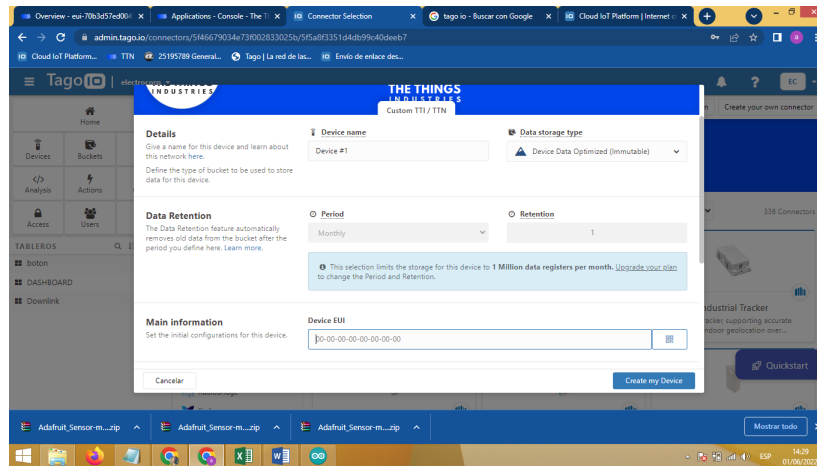


Figura 3.16: Inserción del id de Dragino en la plataforma. Fuente: El autor

A continuación se debe dirigir a Devices ->Generate Information->Generate Tokens, de la misma manera que se lo muestra en la Figura 3.17.

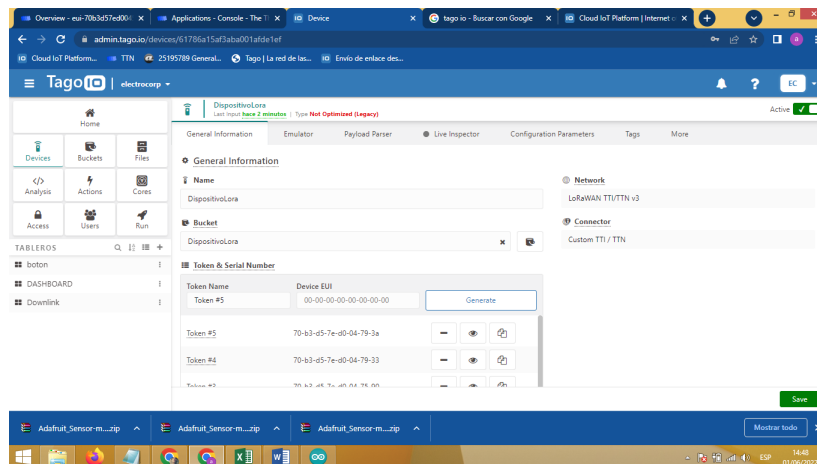


Figura 3.17: Ruta para la generación del Token ID. Fuente: El autor

En Token Name se debe el nombre y EUI de nuestro nodo lora, una vez ingresado los datos seleccionamos la opción Generate.

Nota: una vez agregado los datos anteriormente, encendemos nuestro nodo lora para que se realice la conexión y empiece a enviar datos.

Para visualizar si existen datos recibidos se debe dirigir a la opción Live Inspector para así mostrar los datos conseguidos, tal como se lo muestra en la Figura 3.18.

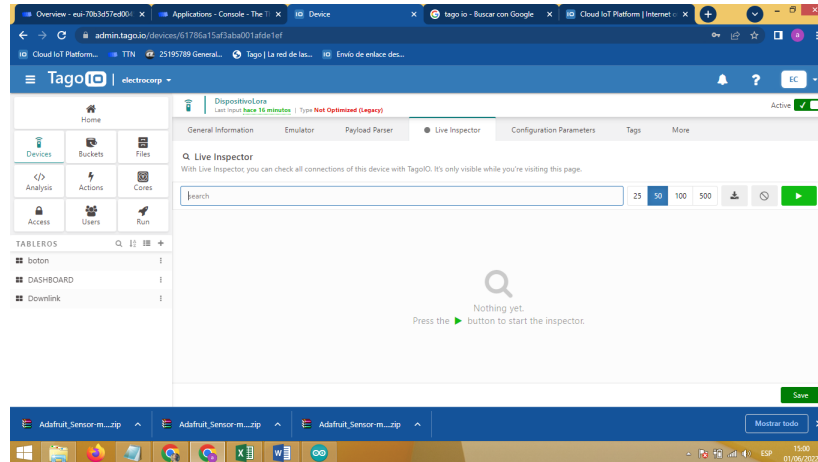


Figura 3.18: Live Inspector del Dispositivo Lora. Fuente: El autor

Una vez ya registrado los datos se deberá proceder a a diseñar el Dashboard y para esto se deberá seleccionar la opción Tableros, tal como se lo indica en la Figura 3.19

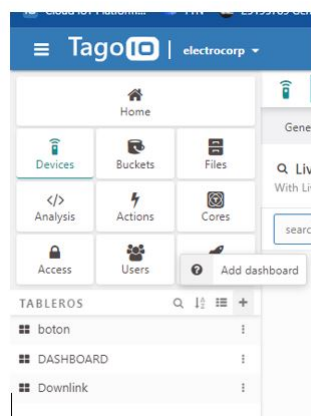


Figura 3.19: Selección de la opción de Tablero. Fuente: El autor

Ya seleccionada la opción de Dashboard se procede a agregarle un nombre

y seleccionar la opción Normal en el tipo de tablero, del mismo modo que se lo indica en la Figura 3.20.

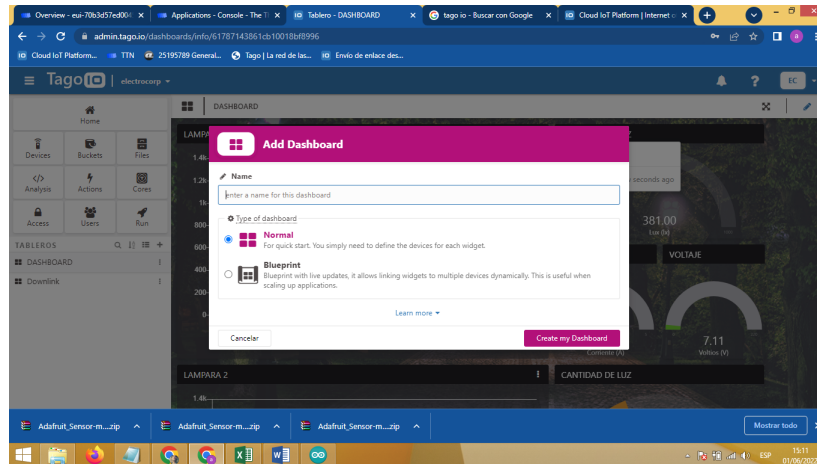


Figura 3.20: Selección del nombre y tipo del tablero. Fuente: El autor

La plataforma como tal brinda una amplia gama de elementos que se podrán incorporar en el tablero, tal como se lo indica en la Figura 3.22.

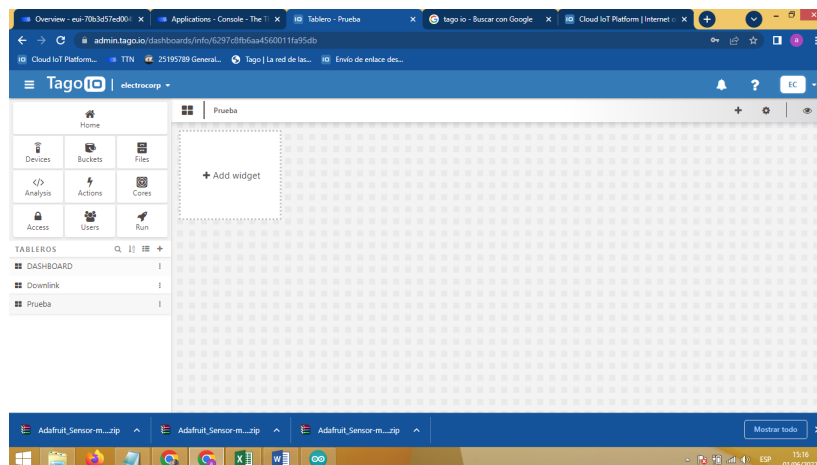


Figura 3.21: Incorporación de Widgets. Fuente: El autor

Entre los elementos o widgets que incorpora esta plataforma se destacan:

- Map

- Display
- Card
- Keypad
- Dynamic Table
- Solid
- Line
- Area

Para el ejemplo su utilizaron el Solid y Line. Para el caso de Solid se debe realizar las siguientes configuraciones reflejada en la Figura 3.22, Figura 3.23, Figura 3.24 y Figura 3.25.

1. Seleccionar el widget Solid.

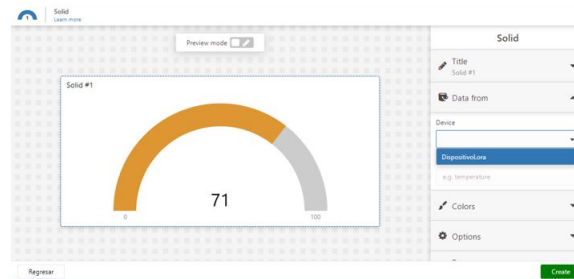


Figura 3.22: Incorporación del Widget Solid. Fuente: El autor

2. Una vez seleccionado se escoge el dispositivo con el que se quiera vincular.

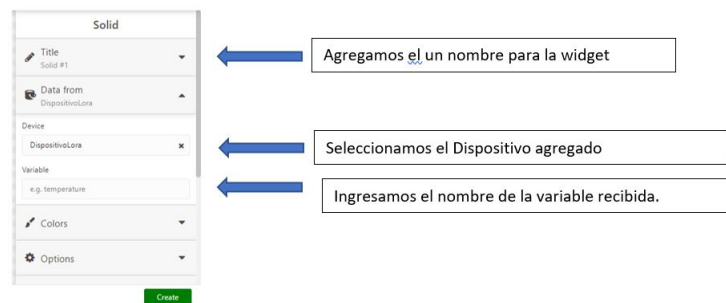


Figura 3.23: Vinculación al Dispositivo agregado. Fuente: El autor

3. Ya para personalizar el elemento se podrá cambiar las matices de colores las cuales podrán ser escogidas desde el pestaña de Colores.

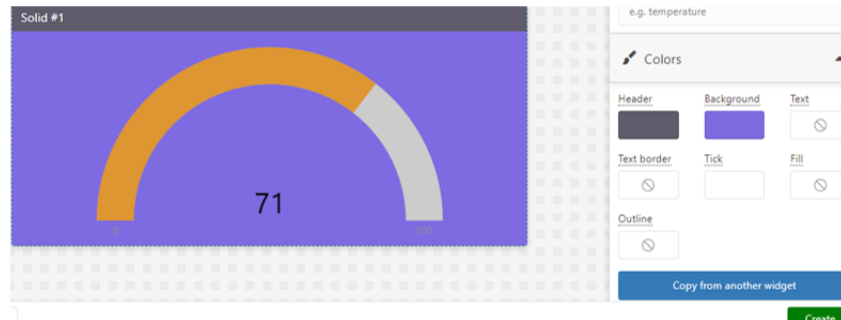


Figura 3.24: Personalización de colores. Fuente: El autor

4. Y finalmente se podrá configurar el rango de valores con los que se desee trabajar.

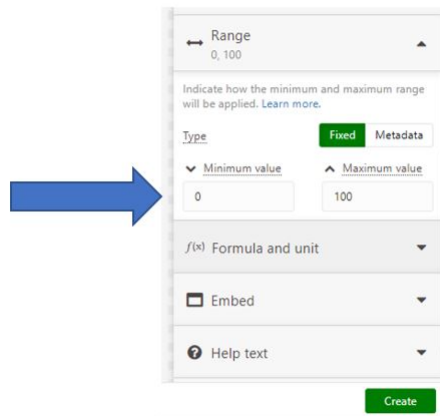


Figura 3.25: Personalización de rangos de valores. Fuente: El autor

En cuanto al widget Line se deberá realizar los pasos reflejados en la Figura 3.26 y Figura 3.27

1. Seleccionar el widget Line.

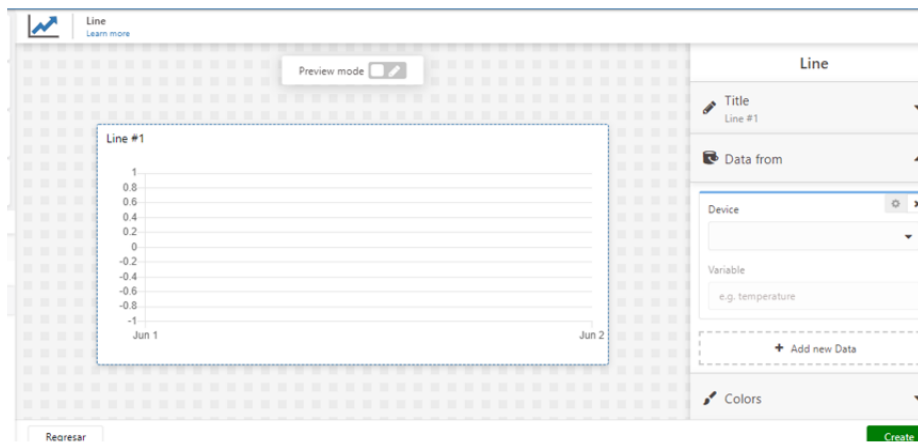


Figura 3.26: Incorporación del Widget Line. Fuente: El autor

- Una vez seleccionado se escoge el dispositivo con el que se quiera vincular, otro punto a agregar es que este widget te permite agregar las variables que sean necesarias.

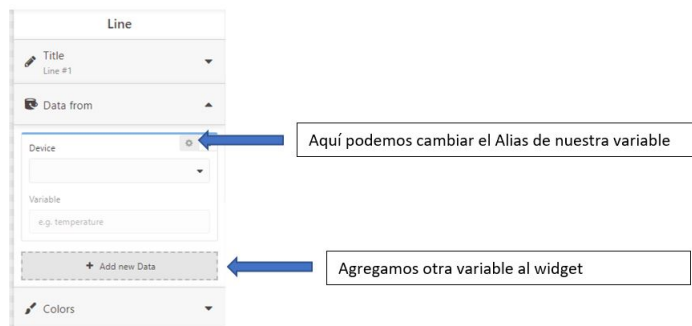


Figura 3.27: Vinculación al Dispositivo agregado. Fuente: El autor

Incorporación de Estados de Alarmas

Para incorporar estados de alarmas se debe realizar los pasos visualizados en la Figura 3.28 y en la Figura 3.29

- Seleccionar el widget Icon.

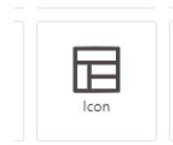


Figura 3.28: Widget Icon. Fuente: El autor

2. Ya seleccionado se debe personalizar el elementos de acuerdo a los campos que se necesitan

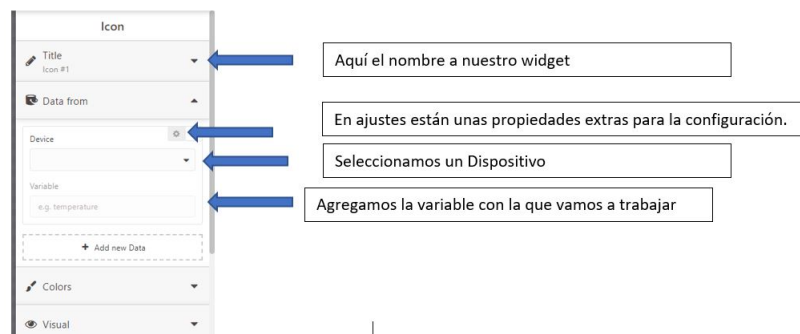


Figura 3.29: Personalización de Widget. Fuente: El autor

3. Y finalmente este elemento brinda la posibilidad de agregar las condiciones e iconos que se requieran, como lo expone la Figura 3.30 y la Figura 3.31.

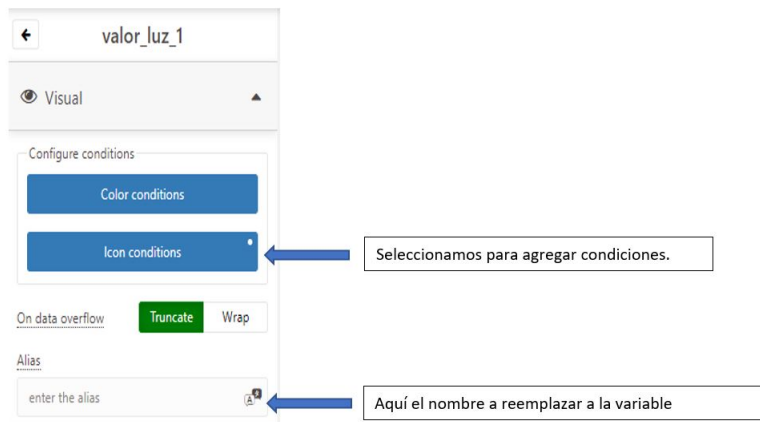


Figura 3.30: Selección de iconos. Fuente: El autor

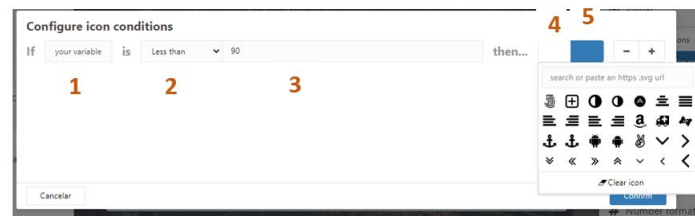


Figura 3.31: Inserción de Condiciones. Fuente: El autor

3.1.3. Diseño del Fuzzy

Fuzzy Control es una librería de estructuras de datos y funciones para construir matrices difusas, en la cual en el presente proyecto se evalúa las variables del controlador difuso para un sistema de iluminación en el software Arduino. En la siguiente figura se define los conjuntos de variables, tanto como un universo de discurso de la variable de entrada y uno para la variable de salida, como se lo refleja en la Figura 3.32 y la Figura 3.33.

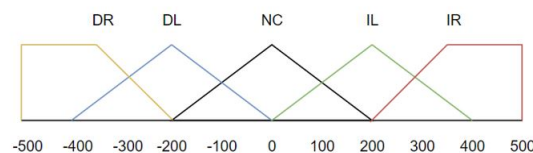


Figura 3.32: Universo de discurso del error. Fuente: El autor

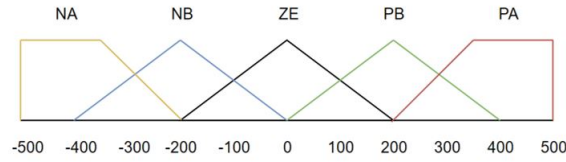


Figura 3.33: Universo de discurso del PWM. Fuente: El autor

Y para su uso se procede a incluir la librería con su respectiva objeto tipo fuzzy llamado “control fuzzy”, en la segunda parte del código se definen los conjuntos de variables del universo de discurso en lo cual se lo representa como USAL (PWM) y UIN (ERROR) al final se definen los conjuntos difusos especificando sus límites y colocando una “T” si son triangulares o una “R” si son trapezoidales. Tal como se lo muestra en la Figura 3.34:

```
Control_fuzzy
// CONTROL DIFUSO PARA SISTEMA DE ILUMINACION
const int LED = 3; // Indicando que el led se encuentra en el pin 3
#include <fuzzy.h> // Incluye la libreria
fuzzy controlfuzzy; // Crea un objeto de tipo fuzzy.

// DEFINICION DE CONJUNTOS DE VARIABLES
// Definimos conjunto para variable de entrada
float USAL[] = {-44, 44}; // Definimos el universo de discurso (rango) de la variable salida (PWM)
float UIN[] = {-500, 500}; // Definimos universo de discurso de la variable de entrada (ERROR)

// Definimos los conjuntos difusos, colocando "T" si es triangular, o "R" si es trapezoidal
// CONJUNTO DIFUSO DE ENTRADA
float NB[] = {-500, -500, -350, -200, 'R'}; // Error Negativo Alto
float NB[] = {-400, -200, 0, 'T'}; // Error Negativo Bajo
float ZE[] = {-200, 0, 200, 'T'}; // Error Cero
float PB[] = {0, 200, 400, 'T'}; // Error Positivo Bajo
float PA[] = {200, 350, 500, 500, 'R'}; // Error Positivo Alto

// CONJUNTO DIFUSO DE SALIDA
float DR[] = {-44, -44, -40, -32, 'R'}; // Decremento Rapido
float DL[] = {-40, -32, 0, 'T'}; // Decremento Lento
float NC[] = {-32, 0, 32, 'T'}; // NO CONTROL
float IL[] = {0, 32, 40, 'T'}; // Incremento Lento
float IR[] = {32, 40, 44, 44, 'R'}; // Incremento Rapido
float paso = 2; // Se define la cantidad minima de variación para los numeros del conjunto de salida
float stepsize = 700, lum=0, error;
const int tam = controlfuzzy.calc_size(USAL, paso); // Calculamos el tamaño que tendrá el vector del conjunto de salida con universo salida y "paso"
```

Figura 3.34: Universo de discurso del PWM. Fuente: El autor

Asimismo, se procede a verificar la inferencia y para ello se debe relacionar el conjunto de entrada con el de salida, tal como se lo indica en la Tabla 3.2.

Tabla 3.2: Tabla de Entrada y Salida

ENTRADA	SALIDA
NA	DR
NB	DL
ZE	NC
PB	IL
PA	IR

Nota. La tabla muestra las entradas y salidas definidas. Fuente: El autor

Y este proceso se lo realiza mediante el paradigma REGLA SIMPLE,

la cual recibe como parámetros (Conjunto entrada, Universo de entrada, variable a evaluar, conjunto salida, universo salida, vector salida, tamaño vector salida).

Ya realizada la inferencia se debe calcular el valor para defusificar con la función “defusi” y lo guardamos en la variable “res”, este valor se añadirá a un valor de 127, dicho valor en el PWM mantiene al valor medido en el puerto A0 en aproximadamente 700 (nuestro setpoint). Posteriormente se escribe este valor en el PWM del pin 3 en el Arduino, como lo indica en la Figura 3.35.

[illegible]

Figura 3.35: Proceso de Defusicación. Fuente: El autor

A continuación se debe proceder a leer la luminancia y esto se lo realiza mediante el divisor de de voltaje LDR, tal como se lo presenta en la Figura 3.36.

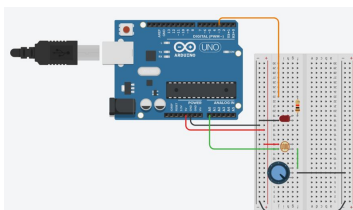


Figura 3.36: Diagrama de conexión. Fuente: El autor

Una vez obtenido el modelo de la planta **Prueba del controlador Fuzzy**, dentro de la sección 5, se procede a realizar el diseño del controlador Fuzzy, para que la planta trabaje en lazo cerrado, se usa en este proceso simulaciones usando el simulador simulink de matlab y el diagrama de bloques reflejado en la Figura 4.5. En esta se puede apreciar que se tiene tres grandes etapas:

- La parte del controlador fuzzy a la cual se le ingresa el error y la derivada del error con el objetivo de tener información en el controlador

de que tan cerca esta el valor de la salida de la planta, y la velocidad con la cual esta cambiando el error.

- Obtención del modelo de la planta.
- Y la determinación de los saturadores de salida del controlador FUZZY.

Gracias a estas etapas se determinó la necesidad de colocar una constante en la señal de error, ya que así se podrá modificar las funciones de membresía desde el exterior del controlador, tanto para el error como para derivada del error y del mismo modo en la salida de controlador. Como se lo muestra en el diagrama reflejado en la Figura 3.37

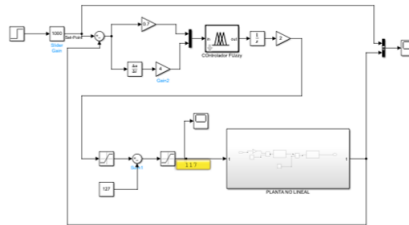


Figura 3.37: Diagrama de conexión. Fuente: El autor

En cuanto a la configuración del controlador Fuzzy se estima que este controlador deberá tener dos entradas, las cuales corresponden al error, a la derivada del error y a la salida de controlador, como se lo detalla en la Figura 3.38

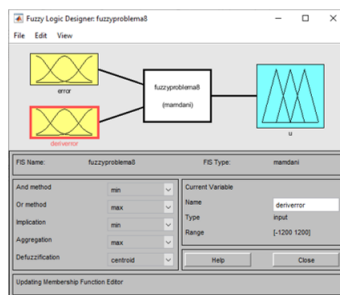


Figura 3.38: Controlador del Fuzzy. Fuente: El autor

Las funciones de membresía de la entrada del error son cinco, las cuales corresponden a:

1. Error negativo grande (ENG).
2. Error negativo medio (ENM)
3. Error cero (EZ).
4. Error positivo medio (EPM).
5. Error positivo grande (EPG).
6. Variable de discurso, el cual va desde 1200 a -1200 lo que corresponde al error máximo que puede tener el sistema, como se lo detalla en la Figura 3.39.

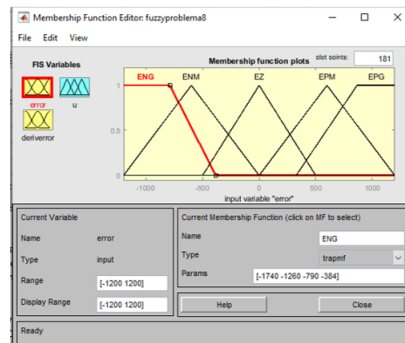


Figura 3.39: Funciones de membresía. Fuente: El autor

Para la derivada del error se ha configurado cinco funciones de membresía las cuales son:

1. Derivada negativa grande (DNG).
2. Derivada negativa media (DNM).
3. Derivada cero (DZ).
4. Derivada positiva media (DPM).
5. Derivada positiva grande (DPG), como se lo indica en la Figura 3.40.

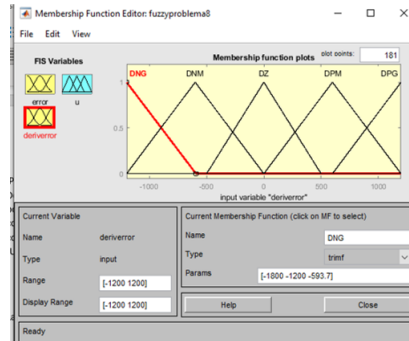


Figura 3.40: Funciones de membresía. Fuente: El autor

Asimismo, las funciones de membresía correspondiente al desfusificador son:

1. Señal de control negativa grande (UCNG).
2. Señal de control negativa media (UCNM).
3. Señal de control cero (UCZ).
4. Señal de control positiva media (UPCM)

Estas señales se pueden observar en la Figura 3.41

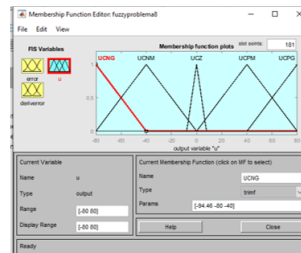


Figura 3.41: Funciones de membresía. Fuente: El autor

Una vez colocado el controlador fuzzy en el simulink con las funciones de membresía mostradas se realizan pruebas de funcionamiento en el simulador las cuales se muestran en la Figura 3.42 en donde se observa que el controlador Fuzzy sigue la referencia estabilizándose en 1000 luxes que es el valor al cual se necesita una eliminación óptima en los parques

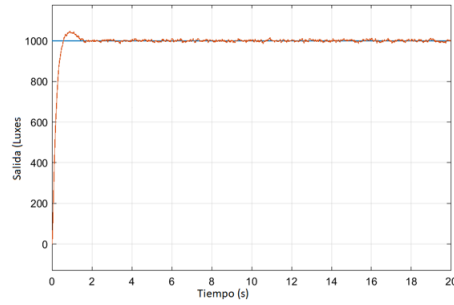


Figura 3.42: Controlador Fuzzy dentro de Simulink. Fuente: El autor

La señal de control que se obtiene para la salida mostrada en la Figura 3.42 se muestra en la Figura 3.43, en esta figura se observa que la acción de control es alta al principio y luego baja para estabilizarse alrededor de los 117.

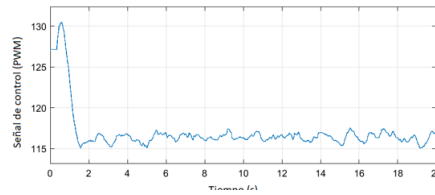


Figura 3.43: Señal de control. Fuente: El autor

Para verificar la efectividad del controlador diseñado se le realizaron pruebas de perturbación a la entrada colocando un función escalón a la entrada de la planta la cual es de 20 PWM 6 segundos después de haberse estabilizado la salida, luego a los 12 segundos se le aplica un escalón negativo de 20 PWM en el diagrama de bloques de la Figura 3.44 se muestra estas perturbaciones.

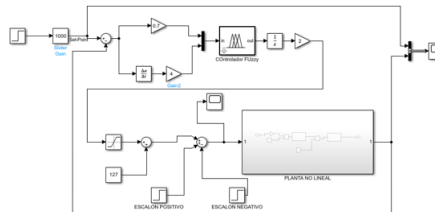


Figura 3.44: Señal de control. Fuente: El autor

Las gráficas de la salida y entrada de la planta ante estas perturbaciones se muestran en la Figura 3.45 y en la Figura 3.46 en donde se puede apreciar que la salida de la planta se ve afectada por esta perturbación, pero el controlador fuzzy logra regresar la variable de salida al setpoint establecido de 1000 luxes, en la señal de control (Figura 3.47) se observa una disminución de la acción de control en el momento que hay una perturbaciones positiva en la entrada de la planta, pero luego el control logra estabilizarla, así mismo en la perturbación negativa en la entrada se observa que hay una aumento en la acción de control (Figura 3.47), con esto el controlador logra estabilizarla en el valor que necesita en la entrada la planta para mantener el setpoint establecido .

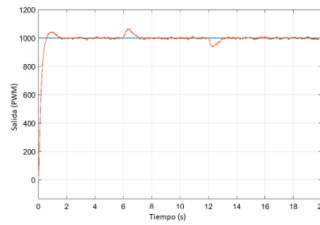


Figura 3.45: Señal de control. Fuente: El autor

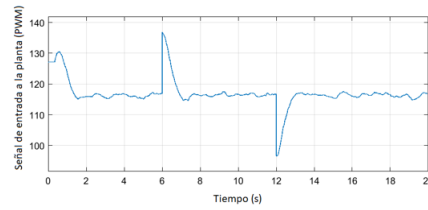


Figura 3.46: Señal de control. Fuente: El autor

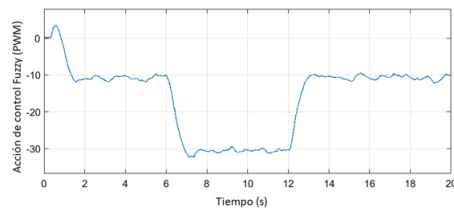


Figura 3.47: Señal de control. Fuente: El autor

De la misma manera se realizaron pruebas de perturbaciones en la salida de la planta (Figura 3.48) con dos escalones, colocadas una a los 6 segundos de haber iniciado la planta con magnitud de 100 luxes, y otra escalón a los 12 segundos con una magnitud de 200 luxes, esta segunda señal en el sumador del diagrama de bloques resta para con esta provocar una disminución en la salida, en la Figura 3.49 y en la Figura 3.50. De esta manera se muestra la salida de la planta en dónde se observa que el controlador logra llevar la salida del sistema al referencial y la acción de control en el momento que hay una subida de la señal de salida de la planta disminuye la acción de control y en el caso donde se provoca un escalón negativo de perturbación la acción de control aumenta para provocar una compensación en la salida.

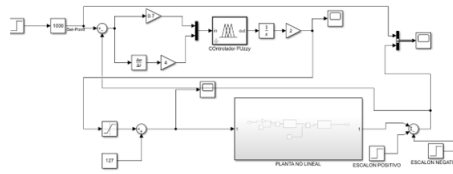


Figura 3.48: Señal de control. Fuente: El autor

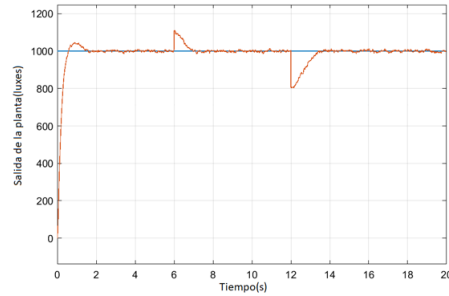


Figura 3.49: Señal de control. Fuente: El autor

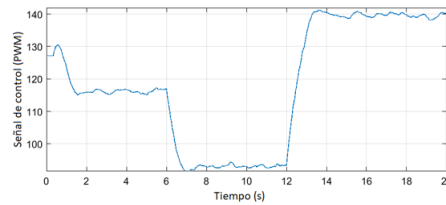


Figura 3.50: Señal de control. Fuente: El autor

3.1.4. Diseño de la tarjeta PCB

Se ha diseñado el circuito electrónico para una alimentación de 12V dc, este voltaje nos va a servir para alimentar directamente el arduino nano. La siguiente es una etapa para estabilizar un voltaje de 5v a través de un regulador 7805, el mismo que se utilizará para alimentar el modulo lora, ya que su alimentación externa solo permite 5vdc. Cabe destacar que todo el circuito electrónico está protegido por medio de un fusible de 1.5 Amperios y también se ha colocado un diodo de protección para polaridad inversa, el cual servirá como medida de contingencia en caso de una conexión equivocada de la alimentación de la tarjeta. Adicionalmente se ha colocado un led de visualización que nos indicará cuando la tarjeta sea alimentada por el voltaje, como lo muestra la Figura 3.51

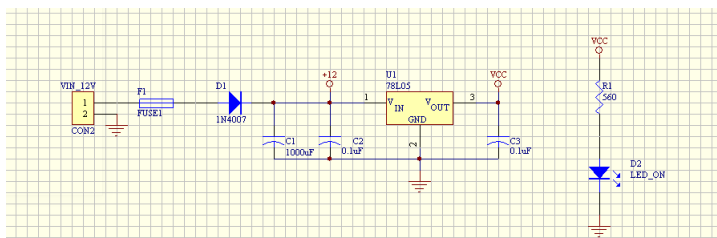


Figura 3.51: Diagrama del circuito. Fuente: El autor

Para el sensor de corriente se utiliza un etapa de acoplamiento de la señal, conformada por dos resistencias en serie de 470k y un capacitor de 10uf, esto permite crear un offset de 2.5v para que el rango de voltaje que devuelve el sensor este de 0v a 5v y pueda ser leída desde el conversor analógico digital del microcontrolador del arduino nano.

El sensor de voltaje nos devuelve una señal ya acoplada, por ello se conecta directamente a un puerto analógico a través de unas borneras de conexión. Así mismo el sensor de luminosidad nos devuelve una señal acoplada en el mismo modulo por lo tanto la conexión será directa a un puerto analógico, del mismo modo que se ve expuesto en la Figura 3.52

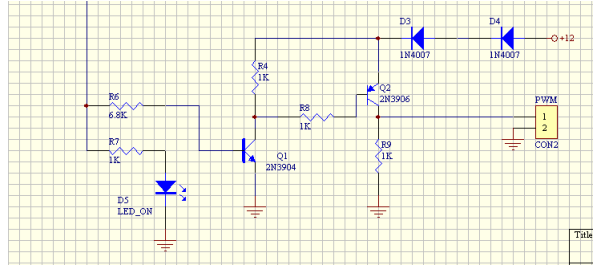


Figura 3.52: Diagrama del circuito controlador de la lámpara. Fuente: El autor

La salida PWM generada por el arduino nano es una señal de 0 a 5V, pero el controlador de la lámpara acepta una señal de control pwm de 0 a 10V, para ello se utiliza el siguiente circuito de acoplamiento en dónde se debe utilizar dos transistores complementarios: el NPN 2n3904 y el PNP 2n3906 en configuración corte y saturación. El transistor NPN recibe la señal de 0 a 5V del arduino nano y activa el transistor PNP el mismo que esta alimentado con 10V aproximadamente en su pin emisor, para lograr el voltaje anteriormente mencionado utilizamos dos diodos en serie. Con esta configuración logramos una salida PWM de 0 a 10v la misma que nos servirá para manejar el controlador de la lámpara, como lo indica la Figura 3.53

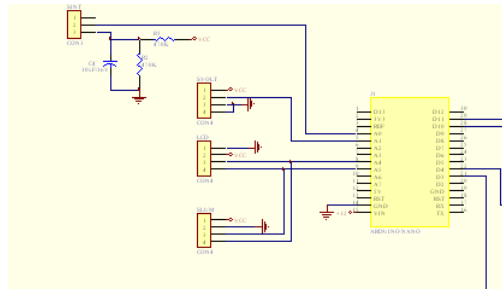


Figura 3.53: Conversor analógico digital del microcontrolador. Fuente: El autor

3.2. Implementación del proyecto

3.2.1. Implementación de la red LoRa

El LG308 3.54 es un gateway de código abierto que permite unir la red inalámbrica LoRa a una red IP a través de WiFi, Ethernet, red celular 3G ó

4G. Asimismo, le facilita a los usuarios poder enviar datos y alcanzar rangos extremadamente largos a velocidades destacables.



Figura 3.54: Gateway LG308. Fuente: El autor

Conexión a través de WiFi

Al iniciar LG308 por primera vez, se genera automáticamente una conexión a través de la red Wifi no segura dragino-xxxxx, pero en caso de que el equipo cuente con un firmware más actualizado la contraseña por defecto es dragino+dragino y una vez conectada la pc tendrá como puerta de enlace por defecto la dirección: 10.130.1.1, tal como se lo indica en la Figura 3.55



Figura 3.55: Conexión a través de WiFi Fuente: El autor

Conexión a través del puerto WAN con DHCP IP desde el enrutador.

Este dispositivo también brinda la posibilidad de conectarse al enrutador a través del puerto WAN. En este caso el LG308 obtendrá la dirección IP

de su enrutador, la cual se la puede encontrar a través del comando `ipconfig` desde el terminal del pc, como se lo muestra en la Figura 3.56.

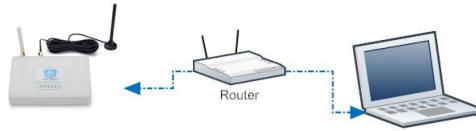


Figura 3.56: Conexión por WAN. Fuente: El autor

Configuración de Dragino

Una vez ya conectado el equipo ya sea a través del modo WiFi o por Wan se procederá a configurarlo y para se debe ingresar a cualquier navegador la dirección ip del Gateway más el puerto 8000, es decir, `http://direcciónip:8000`, tal como se lo indica en la Figura 3.57.



Figura 3.57: Ingreso a la plataforma. Fuente: El autor

Ya ingresada la dirección se deberá utilizar las siguientes credenciales (Figura 3.58):

- Username: root
- Password: Drago

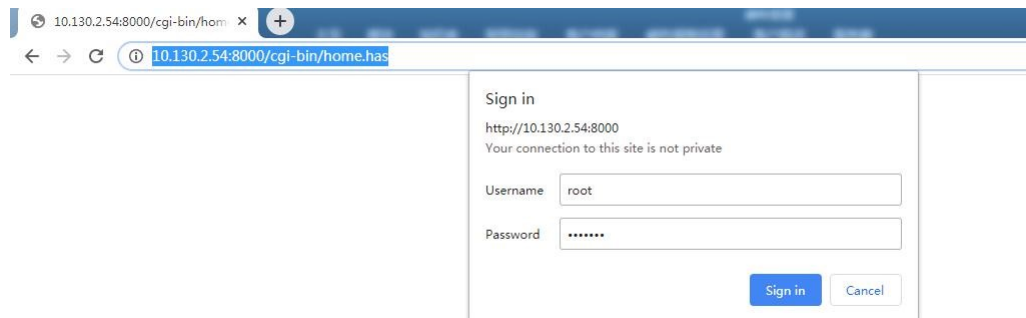


Figura 3.58: Ingreso de las credenciales. Fuente: El autor

Ahora, para la configuración del Wifi Cliente se deberá dirigir a la siguiente ruta: Sistema -> WiFi -> Configuración del cliente WAN WiFi , como se lo indica en la Figura 3.59.

The screenshot shows the 'WiFi' configuration page. The 'WiFi WAN Client Settings' section is highlighted with a green border. It includes the following fields:

- Radio Settings:** Channel (1-11) set to 11, Tx Power (0-18) dBm set to 17.
- WiFi Access Point Settings:**
 - Enable WiFi Access Point: ☒
 - WiFi Name SSID: dragino-1baf44
 - Passphrase (8-32 char): [masked]
 - Encryption: WPA2
- WiFi WAN Client Settings (highlighted):**
 - Enable WiFi WAN Client: ☒
 - Host WiFi SSID: dragino-RD
 - Passphrase: [masked]
 - WiFi Survey: dragino-RD (Ch: 6 Enc)
 - Encryption: WPA/WPA2

Buttons at the bottom: Save&Apply, Cancel, Refresh.

Figura 3.59: Configuración del WiFi Client. Fuente: El autor

Incorporar el Gateway Dragino con el servidor TTN

Para este punto hay que tomar en cuenta que cada LG308 tiene un identificador único, el cual se lo obtiene a través de la página de Dragino, como se lo muestra en la Figura 3.60.

The screenshot shows the 'LoRaWAN Configuration' page. The 'Gateway ID' field is highlighted with a red box. The page includes the following sections:

- General Settings:**
 - Email: dragino-1e9674@dragino.com
 - Gateway ID: a8d0411e96744154 (highlighted)
- Primary LoRaWAN Server:**
 - Service Provider: The Things Network V3
 - Server Address: eu1.cloud.thethings.network
 - Uplink Port: 1700
 - Downlink Port: 1700
- Packet Filter:**
 - Fport Filter: 0
 - DevAddr Filter: 0

Buttons at the bottom: Save&Apply, Cancel.

Figura 3.60: Obtención del Gateway ID. Fuente: El autor

Registro del usuario al servidor TTN

Ya obtenido el ID solo queda registrar el equipo al servidor TTN y para esto se debe ingresar al siguiente enlace:

<https://account.thethingsnetwork.org/register>. Una vez registrado y loggeado se deberá elegir el clúster, tal como se lo presenta en la Figura 3.61.

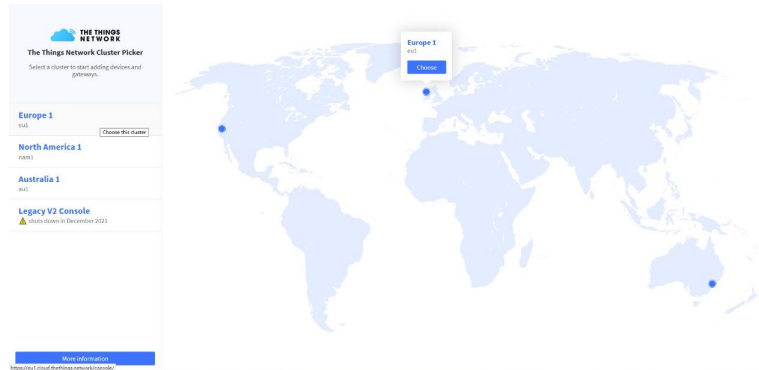


Figura 3.61: Selección del clúster. Fuente: El autor

Un punto a acotar que para elegir el clúster se deben tomar en cuenta ciertos puntos; como el caso de la dirección única perteneciente a cada uno de los gateway, los cuales indicarán a qué región se deberá correlacionar dicho equipo.

- Europa 1: Dirección de servidor de puerta de enlace adecuada: `eu1.cloud.thethings.network`
- Norteamérica 1: Dirección de servidor de puerta de enlace coincidente: `nam1.cloud.thethings.network`
- Australia: Dirección de servidor de puerta de enlace coincidente: `au1.cloud.thethings.network`

Una vez seleccionado el clúster se deberá agregar el Gateway desde la pestaña de overview, tal como se lo indica en la Figura 3.62.

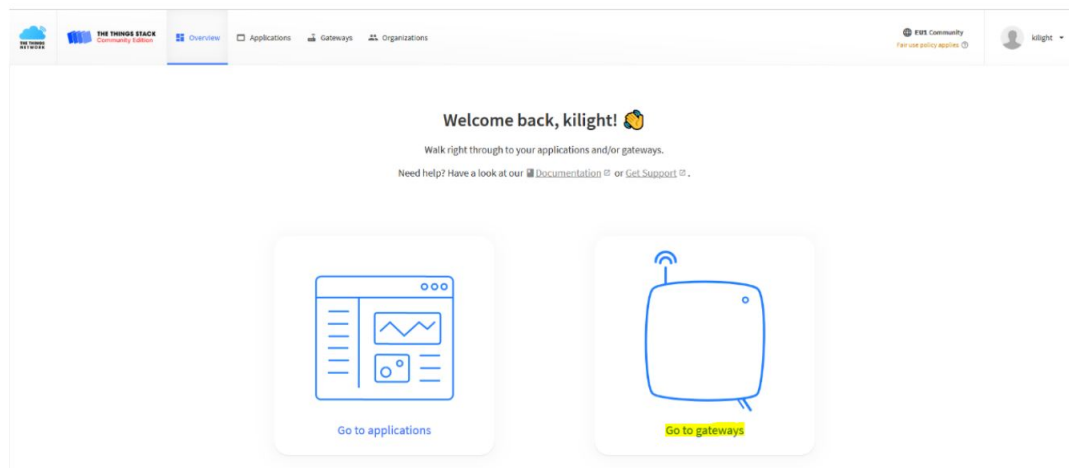


Figura 3.62: Agregar el Gateway. Fuente: El autor

Ya seleccionada esa opción se deberá agregar el ID en la pestaña Gateway EUI, tal como se lo indica posteriormente, como se lo muestra en la Figura 3.63

General settings

Owner *
kilight

Gateway ID *
lpstest-1

Gateway EUI *
AB 48 41 1E 96 74 41 5d

Gateway name
LP58-Gateway

Figura 3.63: Inserción del ID. Fuente: El autor

Después de esto se deberá seleccionar la frecuencia de acuerdo a la región del clúster (Figura 3.64).

Frequency plan
Europe 863-870 MHz (SF12 for RQ2)

Schedule downlink late
☐ Enabled
Enable server-side buffer of downlink messages

Figura 3.64: Selección de frecuencia. Fuente: El autor

Configuración del LG308 para la conexión al TTNv3

Para este punto lo primero que se debe hacer es asegurarse que se cuente con una conexión a Internet activa y posteriormente se deberá seleccionar el proveedor de servidor adecuado, como lo muestra la Figura 3.65.

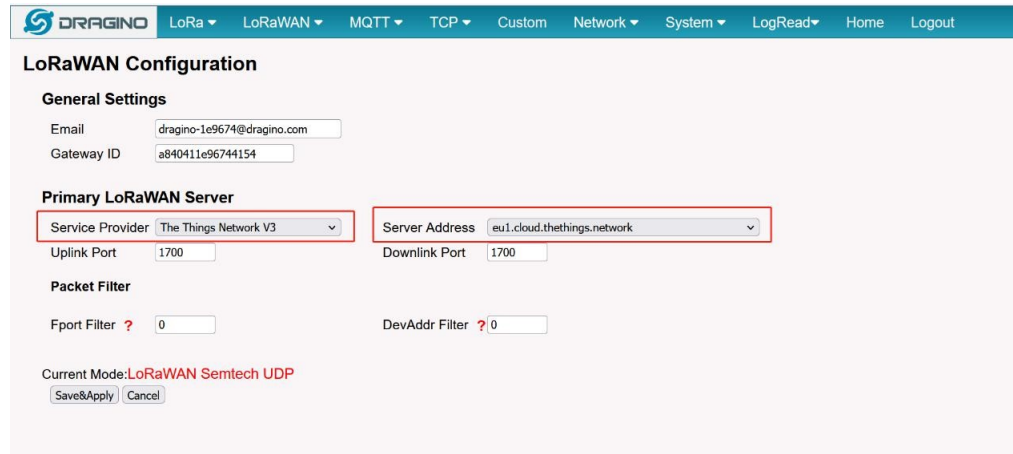


Figura 3.65: Selección del proveedor del servidor adecuado. Fuente: El autor

Y si todos los pasos han sido seguidos a su cabalidad se podrá visualizar que toda la conexión LoRaWAN está lista, como lo indica la Figura 3.66.

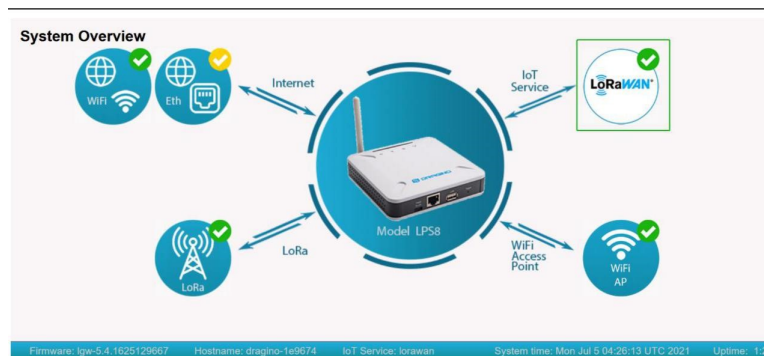


Figura 3.66: Verificación de la conexión LoRaWAN. Fuente: El autor

Agregar un dispositivo final Lorawan

En esta sección se indicará como agregar un dispositivo final LoRaWAN a una red y poder visualizar los datos del sitio web TTN. Y para esto se debe seguir los siguientes pasos:

- Crear una definición de los Dispositivos TTN V3 con las claves OTAA del dispositivo controlador y para esto se requiere definir el prototipo en TTN V3 a través de los siguientes códigos:
 - DEV EUI: código de identificación único para un dispositivo en particular.
 - APP EUI: código de identificación para una aplicación definida en TTN v3.
 - Clave de aplicación: clave única para asegurar las comunicaciones con un dispositivo en particular.
- Posterior a esto seleccione “Agregar Aplicación”, tal como se lo muestra en la Figura 3.67.

The screenshot shows the 'Add application' form in the TTN V3 web interface. The form has the following fields:

- Owner:** A dropdown menu with 'knight' selected.
- Application ID:** A text input field containing 'lor50test'.
- Application name:** A text input field containing 'My new application'.
- Description:** A text area containing 'Description for my new application'.

Below the description field, there is a note: 'Optional application description; can also be used to save notes about the application'. At the bottom of the form is a blue button labeled 'Create application'.

Figura 3.67: Agregar aplicación. Fuente: El autor

- Y a continuación se deberá registrar el dispositivo final, como lo refleja la Figura 3.68.

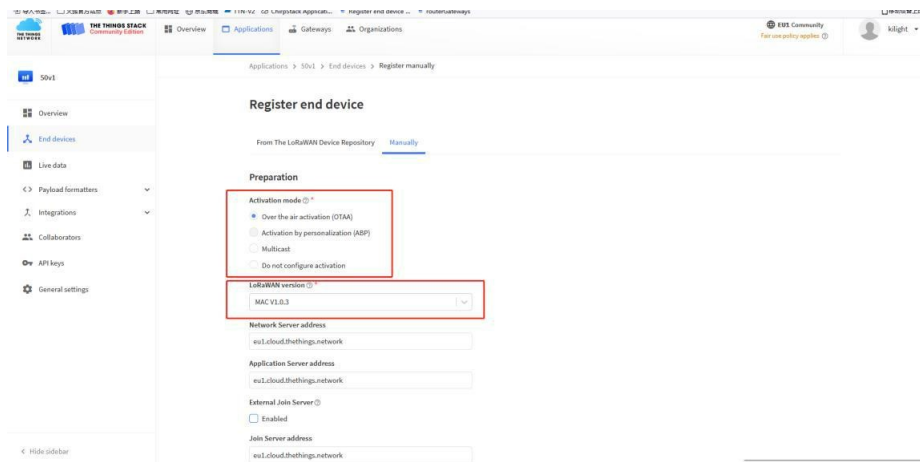


Figura 3.68: Registro del dispositivo final. Fuente: El autor

- Seleccionar modo de activación OTAA, como lo indica la Figura 3.69

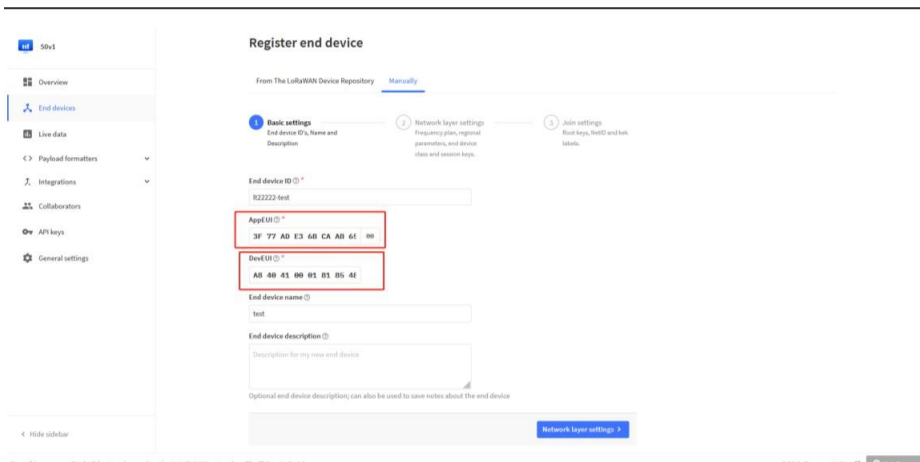


Figura 3.69: Modo de activación OTAA. Fuente: El autor

- Ya seleccionado el modo de activación OTAA se deberá ingresar los ID del Dispositivo Final, AppEUI y DevEUI (Figura 3.70).

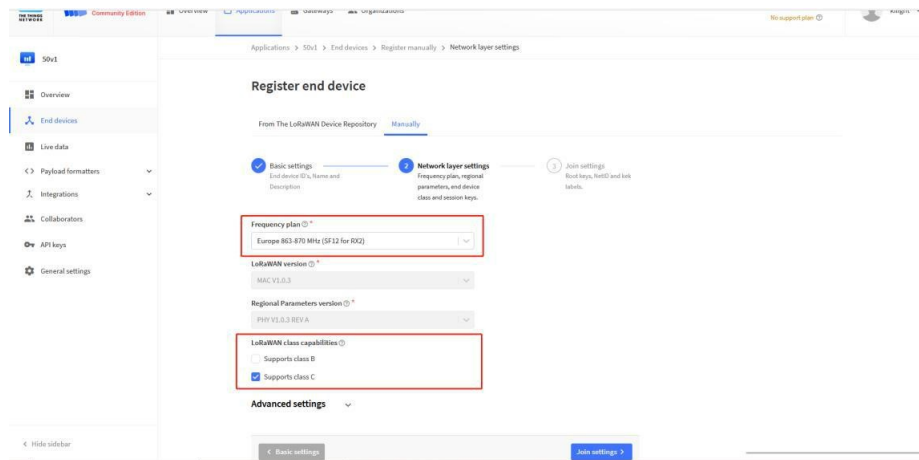


Figura 3.70: Ingreso del ID. Fuente: El autor

- Y finalmente al encender el dispositivo LORA y si los pasos anteriores fueron realizados de manera correcta se deberá unir dicho host a la red TTN y una vez conectado se deberá cargar los mensajes al TTN v3, tal como se lo indica en la Figura 3.71.

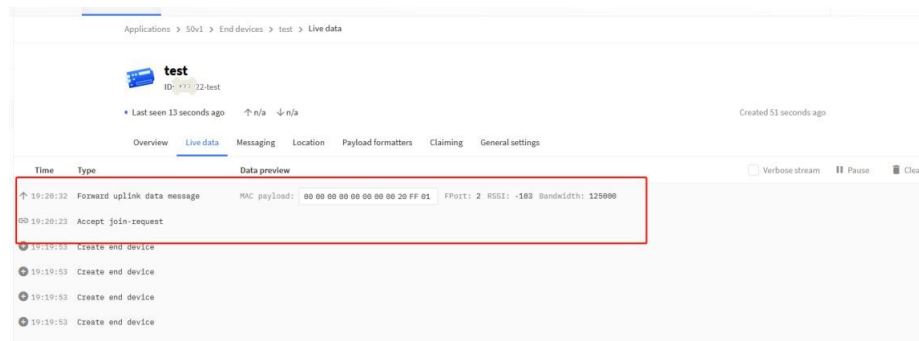


Figura 3.71: Conexión a la red. Fuente: El autor

Configuración del NODO

Para el desarrollo del Nodo Lora a Dragino se deberá seguir los siguientes pasos:

- Se tendrá que declarar las claves de conexión, tanto el devEui, appEui y appKey, como se lo presenta en la Figura 3.72

```

7 static uint8_t devEui[] = { 0x70, 0xB3, 0xD5, 0x7E, 0xD0, 0x04, 0x5C, 0x24 };
8 static uint8_t appEui[] = { 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00 };
9 static uint8_t appKey[] = { 0xA6, 0x79, 0x1B, 0x1D, 0x01, 0x0C, 0x54, 0x8B, 0xDF, 0x4A, 0xFD, 0x03, 0x2D, 0x6D, 0x64, 0x28 };

```

Figura 3.72: Ingreso de tokens. Fuente: El autor

- A continuación, se debe iniciar el programa y conectarse con dragino y para esto se debe enviar las claves como parámetros y en caso de que la conexión haya sido un éxito se lo corroborara a través de un Serial Print, tal como se lo indica en la Figura 3.73.

```

void setup() {
  Serial.begin(115200);

  if (ACTIVE_REGION == LORAMAC_REGION_AU915) {
    LoRaWAN.setSubBand2();
  }
  LoRaWAN.begin(LORAWAN_CLASS, ACTIVE_REGION);
  LoRaWAN.setAdaptiveDR(true);
  while (1) {
    Serial.print("ENVIANDO... ");
    LoRaWAN.joinOTAA(appEui, appKey, devEui);
    if (!LoRaWAN.isJoined()) {
      Serial.println("Envio fallido! Espere 30 segundos");
      lowPowerSleep(30000);
    } else {
      Serial.println("ENVIADO");
      break;
    }
  }
}

```

Figura 3.73: Códec de ingreso de keys. Fuente: El autor

Una vez que el TTN empiece a recibir datos este lo va hacer de forma hexadecimal y por ende, se necesita convertir a decimal y para ello se debe dirigir a la siguiente ruta:

Applications ->End Devices ->seleccionamos un dispositivo
 eui-xxxxxxxxxxxxx ->Payload formatters, como lo indica la Figura 3.74

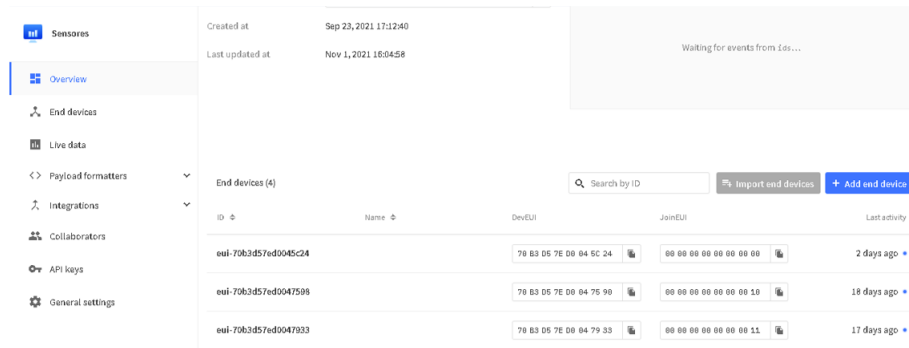


Figura 3.74: Status del equipo. Fuente: El autor

Un punto a destacar es que en formatters types se debe seleccionar JavaScript y posterior se deberá insertar el siguiente código lo que permitirá convertir esta data en decimales, como se lo muestra en la Figura 3.75.

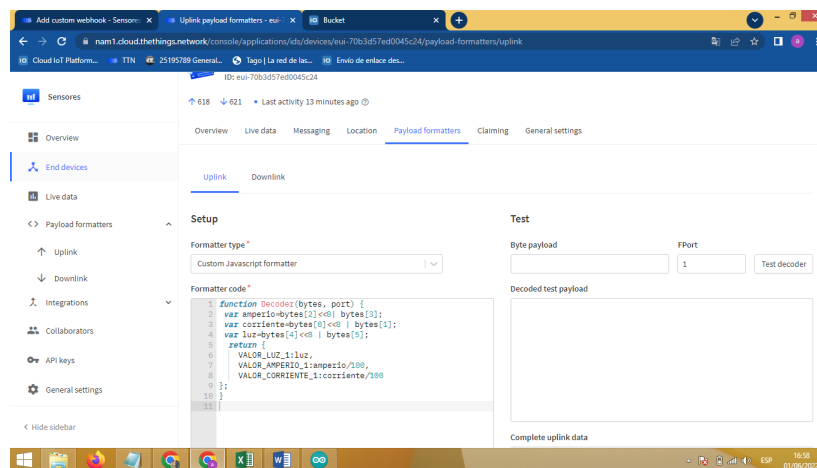


Figura 3.75: Proceso para la conversión a decimal. Fuente: El autor

3.2.2. Implementación de Nexion HMI

Como exordio a esta sección se destaca que Nexion es una solución de tipo HMI (Human Machine Interface), cuya función se centra en proporcionar un interfaz de control y visualización entre humano, máquina y un proceso. Y para implementar este paradigma es necesario descargar el editorNexion, el cual se lo puede conseguir a través del siguiente enlace

<https://nextion.tech/nextion-editor/> y del mismo modo se requiere el uso de algún programa de diseño y en este caso se optó por inkscape.

Cabe destacar que para la realización de este punto se realizaron los siguientes procesos :

1. Se crea un nuevo proyecto en nextion editor y para esto se debe dirigir a Archivos a New y posteriormente se debe seleccionar el nombre del proyecto, como se lo indica en la Figura 3.76.

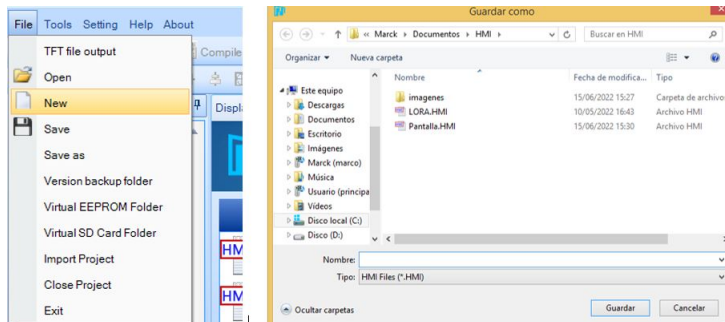


Figura 3.76: Creación de un nuevo archivo. Fuente: El autor

2. Posteriormente se escoge como plantilla base al modelo NX3224T028, como lo muestra en la Figura 3.77.

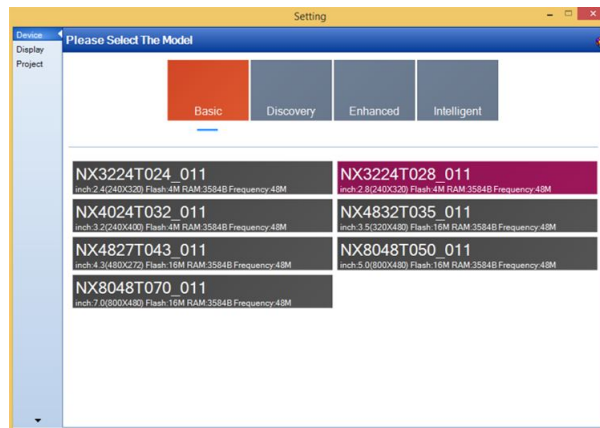


Figura 3.77: Elección de la plantilla. Fuente: El autor

3. Como paso siguiente se debe seleccionar la orientación de la pantalla, el cual se utilizó el de 90, como lo muestra en la Figura 3.78.

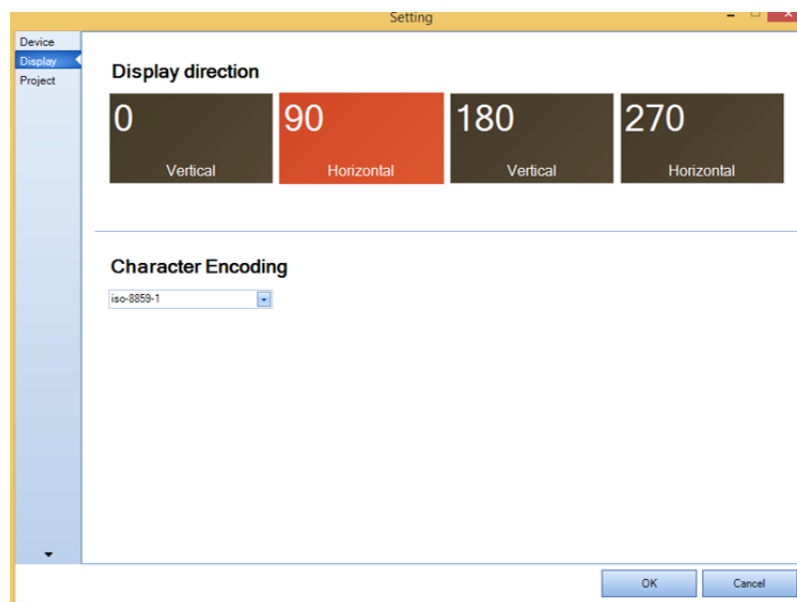


Figura 3.78: Elección de la plantilla. Fuente: El autor

4. Para el diseño lo primero que se necesita es abrir un nuevo proyecto en Inkscape, como lo indica la Figura 3.79

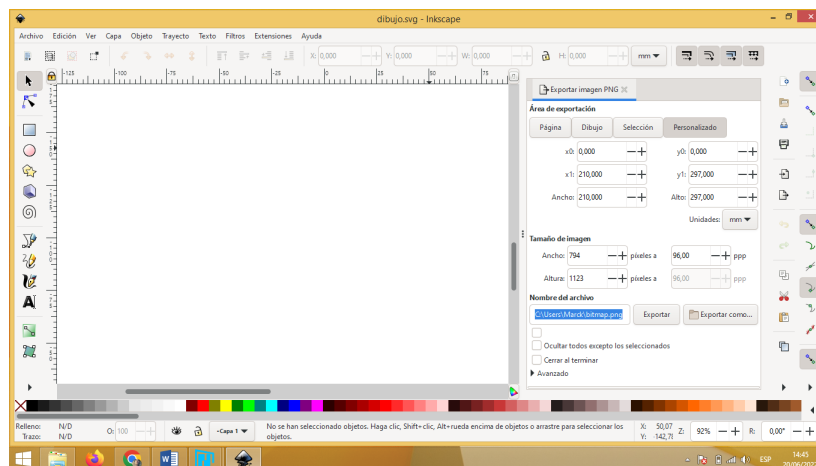


Figura 3.79: Nuevo proyecto en Inkscape. Fuente: El autor

5. Una vez seleccionado se deberá ingresar un fondo y para esto se

puede utilizar una imagen externa o una propia del programa, como lo muestra la Figura 3.80

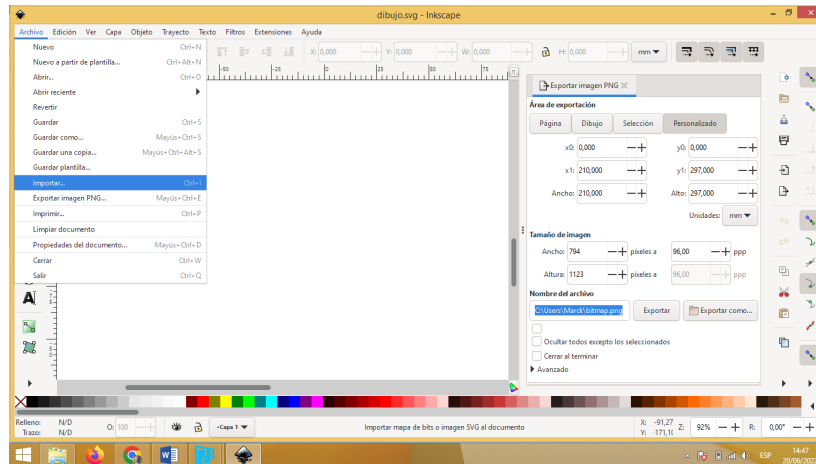


Figura 3.80: Selección del fondo para el proyecto. Fuente: El autor

6. Posteriormente se deberá dimensionar las medidas de la pantalla, para esto se debe ir a propiedades de la ventana y dimensiones, como lo presenta la Figura 3.81

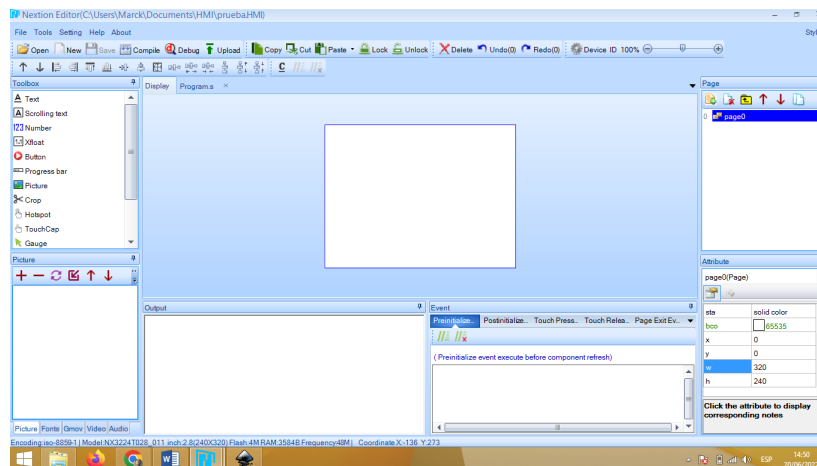


Figura 3.81: Dimensionar la pantalla. Fuente: El autor

7. Una vez dimensionada la pantalla se procede realizar el diseño que se

requiera y exportarla ya sea en formato png o jpg, como lo indica la Figura 3.82 y la Figura 3.83.

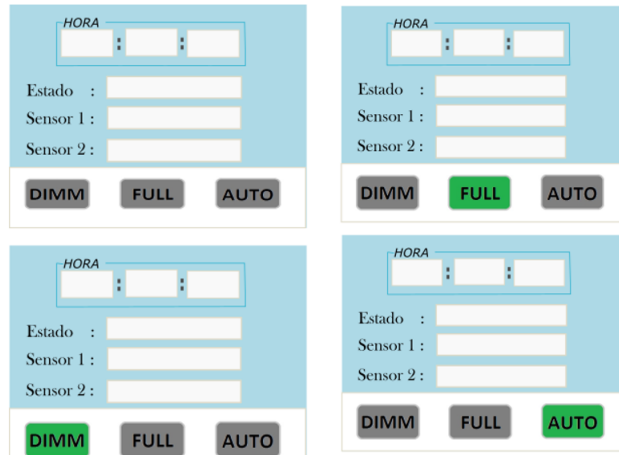


Figura 3.82: Diseños realizados. Fuente: El autor

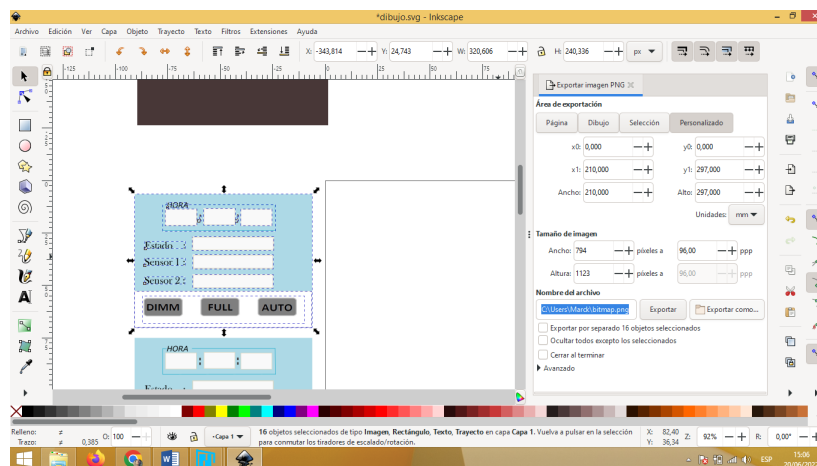


Figura 3.83: Proceso de exportación. Fuente: El autor

8. Ya exportada la imagen, se la tiene que agregar a Nextion Editor y para esto se deberá ir a la opción de Picture y luego al + para así agregar los diseños previamente creados, como se lo visualiza en la Figura 3.84

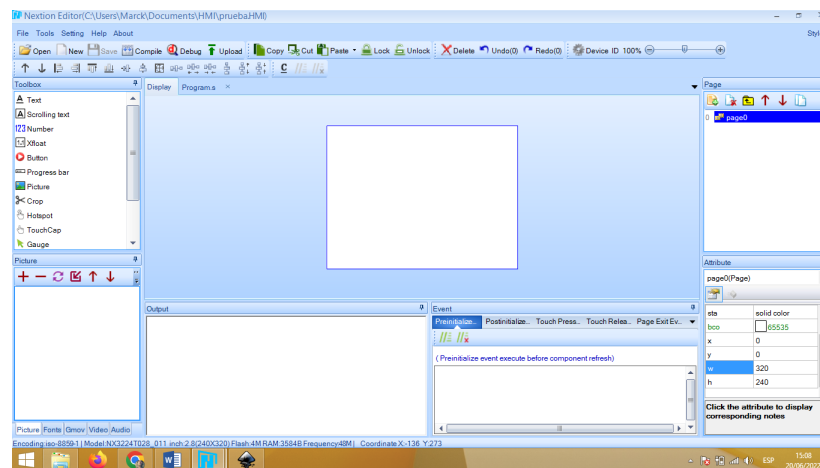


Figura 3.84: Agregar el diseño al editor. Fuente: El autor

9. Posteriormente en propiedades se deberá asignar al campo BCO como Image, ya que esto nos permitirá usar el bosquejo creado como diseño, como lo indica la Figura 3.85.

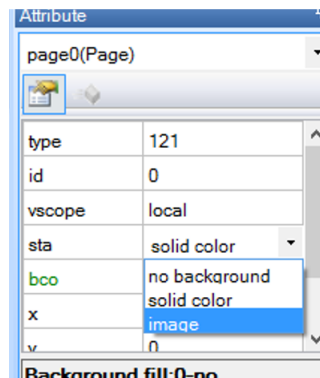


Figura 3.85: Diseño de plantilla. Fuente: El autor

10. Una vez ya definida la plantilla, se sobrescribirá los campos de diseño y para esto se escogerá cajas de textos y botones, tal como se lo presentará en la Figura 3.86 y Figura 3.87.



Figura 3.86: Sobre posición de cajas de textos. Fuente: El autor

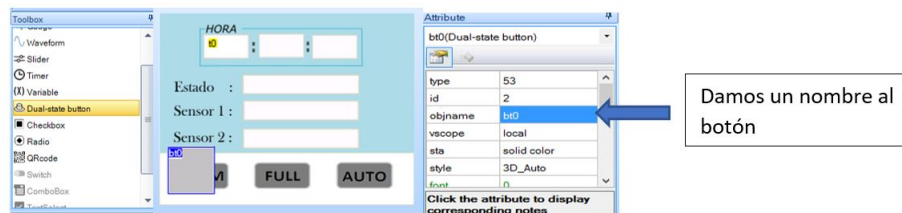


Figura 3.87: Sobre posición de los botones. Fuente: El autor

En cuanto a la codificación para la lectura y envío de datos desde el Arduino al Nextion se llevaron a cabo los siguientes pasos:

1. Agregar la librería Nextion.h
2. Crear instancias tanto para valores Numéricos como para cadenas de texto. Tal como se lo muestra a continuación:

```
NexGauge Nhora = NexGauge(0, 4, "Nhora");
```

```
NexGauge Nminuto = NexGauge(0, 5, "Nminuto");
```

```
NexGauge Nsegundo = NexGauge(0, 6, "Nsegundo");
```

```
NexText Nestado = NexText(0, 7, "Nestado");
```

```
NexText Nsensor1 = NexText(0, 8, "Nsensor1");
```

```
NexText Nsensor2 = NexText(0, 9, "Nsensor2");
```

3. Posteriormente se inicializa la librería dentro de la función setup:

```
void setup()
```

```
Serial.begin(9600);
```

```
nexInit();
```

4. A continuación se crea la función de Leer Datos, la cual no es más que un `Serial.readString`.

```
void leerdatos()
if (Serial.available() >0)
String dato = Serial.readString();
```
5. Una vez ya realizada la codificación en el Arduino se procede llevarla al editor Nextion, y para esto se va a File ->TFT file output y esto permitirá que se genere un archivo .TFT.
6. El archivo generado lo pegamos en una microsd y posteriormente se lo debe colocar en la parte de atrás de la pantalla. Una vez colocada se enciende la pantalla y se transferirá el programa diseñado y finalmente se debe apagar la pantalla, retirar la memoria y volverla a encender para así predeterminedar la funcionalidad diseñada.

3.2.3. Implementación del controlador Fuzzy

Para implementar un controlador Fuzzy es necesario utilizar la librería **fuzzy.h**, y para esto es necesario descargarlo desde el enlace <https://cutt.ly/gLedVrq>, una vez descargado se deben realizar los siguientes pasos:

1. Incluir la librería.

```
#include <fuzzy.h>
```
2. Definir el conjunto de variables de entradas para el controlador Fuzzy.

```
float USAL[]=-64,64;
[PWM] float UIN []=-500,500;
```
3. Definir los conjuntos difusos, para esto se coloca **T** si es triangular o **R** si es trapezoidal.

```
float NA[]=-500,-500,-350,-200,'R';
float NB[] = -400,-200,0,'T';
float ZE[] = -200,0,200,'T';
float PB[] = 0,200,400,'T';
float PA[] = 200,350,500,500,'R';
```

4. Definir los conjuntos de salida.

```
float DR[] = -64,-64,-48,-32,'R'; float DL[] = -48,-32,0,'T'; float NC[]
= -32,0,32,'T'; float IL[] = 0,32,48,'T'; float IR[] = 32,48,64,64,'R';
float paso = 2; float setpoint =700,lum=0,error;
const int tam = controlfuzzy.calc_size(USAL,paso);
```

5. Crear un vector en dónde se almacenarán los números del conjunto de salida.

```
void loop() {
float B[tam];
controlfuzzy.inicio(B,tam);
lum=analogRead(A0);
error=setpoint-lum;
Serial.print(.ERROR:");
Serial.println(error);
(...)}
```

6. Empezar con la inferencia a través, del uso de una regla _simple.

```
controlfuzzy.regla_simple(NA,UIN,error,DR,USAL,B,tam);
controlfuzzy.regla_simple(NB,UIN,error,DL,USAL,B,tam);
controlfuzzy.regla_simple(ZE,UIN,error,NC,USAL,B,tam);
controlfuzzy.regla_simple(PB,UIN,error,IL,USAL,B,tam);
controlfuzzy.regla_simple(PA,UIN,error,IR,USAL,B,tam);
```

7. Emplear una defusificación en el conjunto de salida para así obtener el resultado del sistema fuzzy, para así asignar ese resultado a la variable control.

```
float res = controlfuzzy.defusi(B,USAL,tam); res = 127+res;
analogWrite(LED,res); Serial.println(res); delay(500);
```

3.2.4. Implementación de la placa PCB

Ya esquematizado el diseño de la placa PCB se procede a definir el proceso de ubicación de los componentes, en dónde se destaca la ubicación de los componentes de la manera más ordenada posible para así evitar cruces en las líneas de conexión de la tarjeta, tal como se lo muestra en la Figura 3.88.

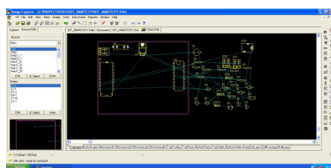


Figura 3.88: Diagrama del circuito controlador de la lámpara. Fuente: El autor

Una vez ubicado los componentes se procede a realizar el enrutamiento de las pistas para así interconectar todos los componentes, para esto se realiza una PCB a doble cara, ya que de esta manera se facilitará la definición de las líneas de conexión, como se lo refleja en la Figura 3.89 .

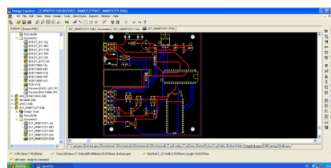


Figura 3.89: Diagrama de la tarjeta en doble cara. Fuente: El autor

Posteriormente, se muestra una simulación del acabado de la tarjeta de PCB, el cual dará las pautas necesarias para llevar a cabo la impresión, como se lo indica en la Figura 3.90.

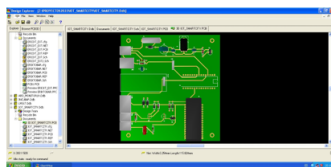


Figura 3.90: Simulación del acabado de la tarjeta. Fuente: El autor

Y como resultado del circuito impreso se obtiene la placa que se presentará en la Figura 3.91.

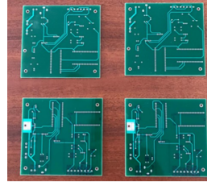


Figura 3.91: Circuito Impreso. Fuente: El autor

3.3. Prueba del proyecto

3.3.1. Prueba del controlador Fuzzy

En cuanto a las pruebas de funcionamiento y gestión del controlador Fuzzy fue necesario determinar el modelo matemático de la planta y para esto fue necesario tomar como datos experimentales los valores obtenidos a través de la red de sensores; cuyos valores se verán reflejados en la Tabla 3.3.

Tabla 3.3: Tabla de extracción de variables

Porcentaje	PWM	Luxes	Tempo
60 %	153	1158	1 Seg
50 %	127	1052	1 Seg
40 %	102	939	1 Seg

Nota. La tabla muestra los datos requeridos para el diseño del modelo matemático. Fuente: El autor

Con los valores reflejados en la tabla 3.3 se construye la ecuación de la recta, la cual se la presentará a continuación

$$Luxes = 4,29 * PWM + 501$$

Y a esta ecuación se le deberá incorporar un valor transitorio el cual será el tiempo que tarda la lámpara para estabilizarse y para esto se debe realizar una prueba en dónde se le asignará como entrada a la lámpara una función escalón de 153PWM, tal como se lo puede presenciar en la Figura 3.92.

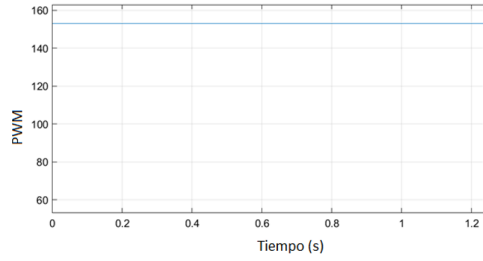


Figura 3.92: Señal de prueba de la entrada de la Lámpara. Fuente: El autor

De manera inmediata, se deberá obtener los índices del luxómetro a relación con el tiempo, ya que se requiere considerar el factor de estabilización (Figura 3.93), tal como se lo mencionó previamente.

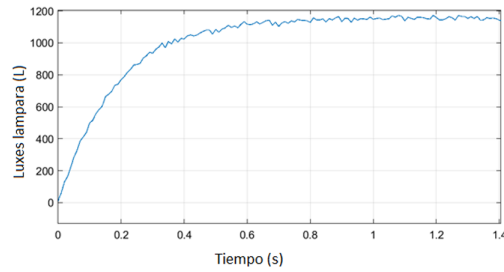


Figura 3.93: Relación de lux con el tiempo. Fuente: El autor

Con la ayuda de la figura 3.93, se obtiene que que la función de transferencia del conjunto lámpara más la tarjeta de fuerza tienen un tiempo de estabilización de 1 segundo con lo cual se le incorporará a la ecuación de la recta anterior la siguiente ecuación diferencial.

$$K\left(\frac{d(luxes)}{dt}\right) + K1 * Luxes = K2 * PWM$$

En dónde K, K1 y K2 al aplicar la transformada de Laplace y obtener la función de la ecuación, se obtiene la siguiente fórmula

$$\frac{Luxes}{PWM} = \left(\frac{k2}{(ks + k1)}\right)$$

Y para obtener las constantes se utilizaron las señales conseguidas previamente y a través, de un serie de prueba y error se obtuvo en el simulador las constantes siguientes:

$$K=0.18$$

$$K1=1$$

$$K2=1$$

Con lo cual se obtiene que la función de transferencia es:

$$\frac{Luxes}{PWM} = \left(\frac{1}{0.18s + 1} \right)$$

Por lo tanto, al realizar el diagrama de bloque del modelo matemático se obtiene el gráfico de la Figura 3.94:

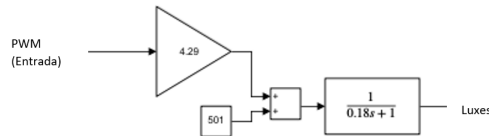


Figura 3.94: Relación de lux con el tiempo. Fuente: El autor

Para validar este modelo es necesario aplicar la señal de entrada a la función escalón de 153 PWM. Obteniendo como modelo de gráfica el diagrama presentado en la Figura 3.95.

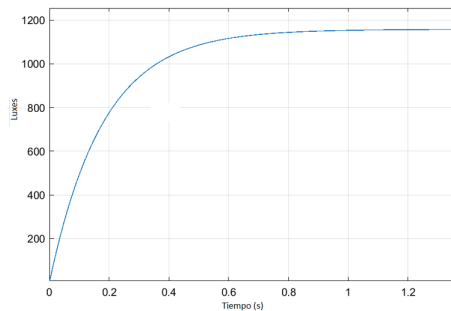


Figura 3.95: Relación de lux con el tiempo. Fuente: El autor

Ya como prueba final del modelo planteado es necesario comprobar los diagramas tanto de la planta real como el del modelo gráfico, las cuales serán presentadas en la Figura 3.96 y en la Figura 3.97

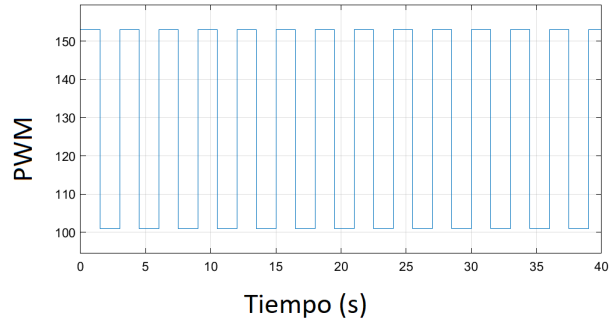


Figura 3.96: Modelo gráfico de la planta real. Fuente: El autor

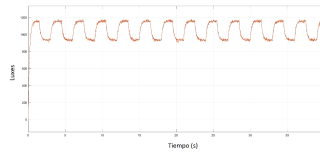


Figura 3.97: Modelo gráfico del modelo planteado. Fuente: El autor

Con el modelo matemático encontrado se procede a calibrar el controlador fuzzy utilizando el simulador Simulink de Matlab para lo cual se utilizó las funciones de membresía que se reflejarán en la Figura 3.98 y Figura 3.99, tanto para la entrada como para la salida del controlador.

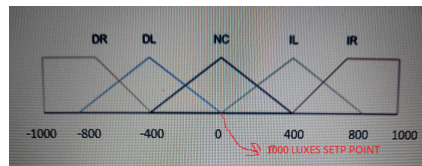


Figura 3.98: Universo de discurso de entrada (error). Fuente: El autor

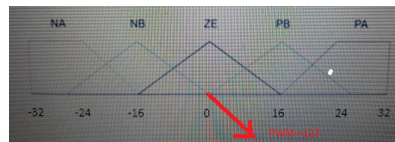


Figura 3.99: Universo de discurso de salida (PWM). Fuente: El autor

Capítulo 4

Resultados y Discusión

Para los resultados de las pruebas de campo se han centrado para comprobar el funcionamiento del controlador Fuzzy y para ello se ha maquetado como un estado inicial cuando la lámpara este totalmente apagado y por ende, se deduce que en este status el sensor muestre un valor de 383.33 con un margen de 611,67, lo que implica que el resultado de desfuzicación es de 18.67, el mismo que al tener un signo positivo se le agregará un valor de 127, dando así como resultado un PWM de salida de 145 lo que permitirá que la lámpara incremente su iluminación para compensar ese margen de error. Tal como se lo muestra en la Figura 4.1.

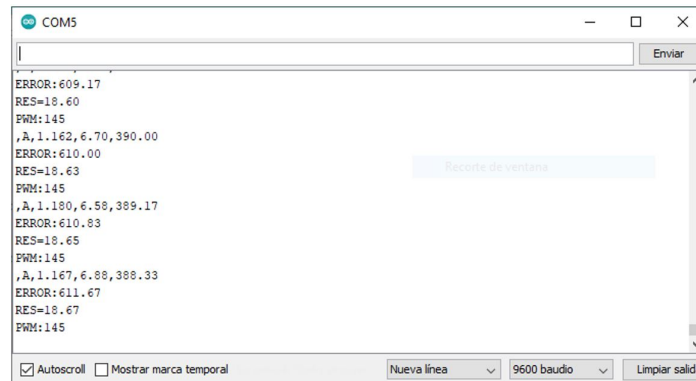


Figura 4.1: Comprobación del Fuzzy. Fuente: El autor

La siguiente comprobación (Figura 4.2) se realiza con un valor de 860.83 presente en el sensor de iluminación, aquí el error es de 139.17, dando como resultado de la desfuzificación un valor de 3.31 que se sumaría al valor inicial

de 127 en el PWM lo que daría como resultado 130 y con este valor se mostraría un incremento leve en la iluminación de la lampara.

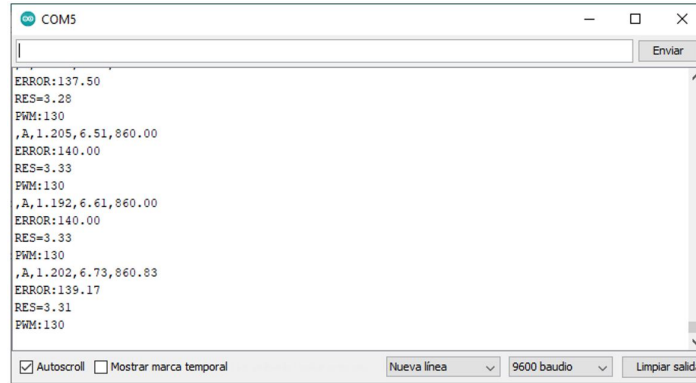


Figura 4.2: Comprobación del Fuzzy. Fuente: El autor

El siguiente resultado se realiza con un valor de 989.17 presente en el sensor, el mismo que es un valor muy cercano al setpoint, por lo tanto el error asignado es de 10.83 y representa un resultado de 0.25 en la desfuzificación. Este valor se adiciona a los 127, dando como resultado los mismo 127. Entonces se comprueba que en este punto el controlador mantiene el valor asignado sin presentar ningún cambio. Como se lo refleja en la Figura 4.3.

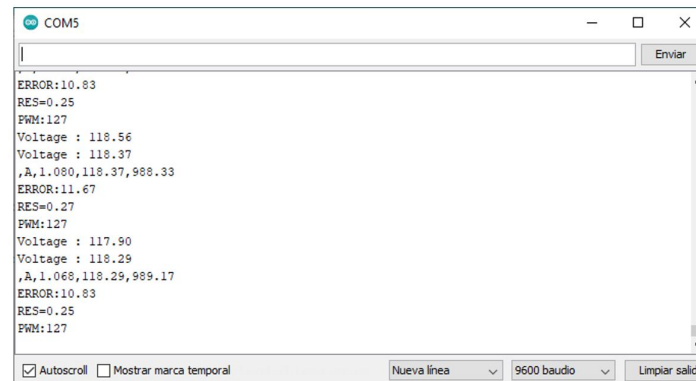
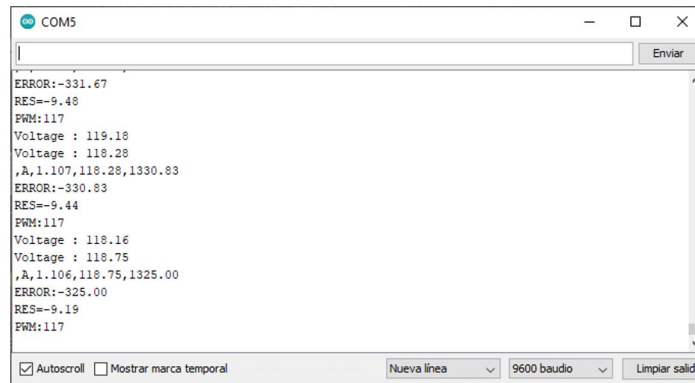


Figura 4.3: Comprobación del Fuzzy. Fuente: El autor

Para complementar los resultados también se consideró como status una sobre iluminación con un valor de 132500 lux, lo que implica que su error sea de 325 lo que corresponde a un resultado de -9.19 lo que sumando los 127 de

margen daría un total de 117 en el PWM, lo que incita que el controlador estaría reduciendo la iluminación para sobrellevar esta anomalía, tal como se lo indica en la Figura 4.4

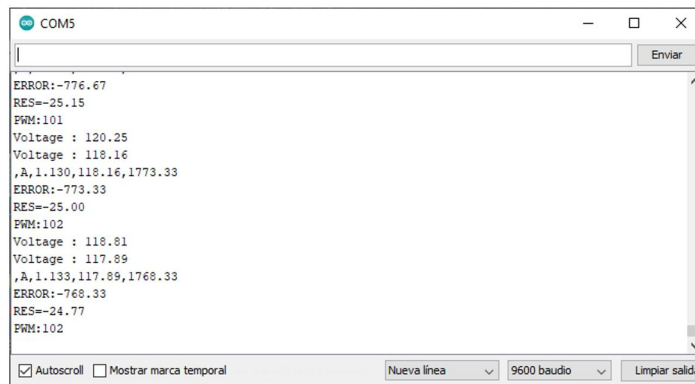


```
COM5
ERROR:-331.67
RES=-9.48
PWM:117
Voltage : 119.18
Voltage : 118.28
,A,1.107,118.28,1330.83
ERROR:-330.83
RES=-9.44
PWM:117
Voltage : 118.16
Voltage : 118.75
,A,1.106,118.75,1325.00
ERROR:-325.00
RES=-9.19
PWM:117
```

Autoscroll ☒ Mostrar marca temporal ☐ Nueva línea 9600 baudio Limpiar salida

Figura 4.4: Comprobación del Fuzzy. Fuente: El autor

Y como última prueba (Figura 4.5) para verificar el funcionamiento del Fuzzy se estimó como escenario una sobre iluminación promedio de 1768.33 lux, dando un error un error de -768.33 y como resultado -24.77, por lo que se debería sumar 127 dando un total de 102 en la salida de PWM, lo que da como resultado un decremento alto de iluminación para así compensar esta situación.



```
COM5
ERROR:-776.67
RES=-25.15
PWM:101
Voltage : 120.25
Voltage : 118.16
,A,1.130,118.16,1773.33
ERROR:-773.33
RES=-25.00
PWM:102
Voltage : 118.81
Voltage : 117.89
,A,1.133,117.89,1768.33
ERROR:-768.33
RES=-24.77
PWM:102
```

Autoscroll ☒ Mostrar marca temporal ☐ Nueva línea 9600 baudio Limpiar salida

Figura 4.5: Comprobación del Fuzzy. Fuente: El autor

Capítulo 5

Conclusiones y Recomendaciones

5.1. Conclusiones

Cumplimiento de Objetivo General

Se desarrolló una plataforma de comunicación basada en IoT y LPQWAN para el control y supervisión los sistemas de alumbrado público en parques y jardines.

Cumplimiento del Objetivo Específico 1

Durante el proceso de investigación se analizaron diversos tipo de tecnología de red LPWAN basados en el paradigma del Internet de las Cosas para luego implementarlas durante el desarrollo de la propuesta, lo que conllevó al uso de tecnologías como el gateway LG308, el cual además de ser compatible con el Arduino permite una puerta de enlace desde una red inalámbrica LoRa a una red IP a través de WiFi, Ethernet, red celular 3G o 4G; lo que permitió el envío de datos al servidor TTN.

Cumplimiento del Objetivo Específico 2

Asimismo, se diseñó un interfaz web intuitivo y amigable en dónde se puede visualizar, monitorear y almacenar los datos en tiempo real y para esto se procedió a vinular la plataforma TTN con Tago IO, el cual nos permite diseñar un dashboard dentro de la plataforma y del mismo modo permite compartir dicha data a usuarios finales. Cabe destacar que además de crear el dashboard por defecto que nos brinda Tago IO se diseño una vista mediante Inkscape dándole así una mayor personalización al interfaz diseñado.

Cumplimiento del Objetivo Específico 3

Para automatizar procesos se desarrolló un sistema de control de Fuzzy,

el cual tiene la capacidad de regular la cantidad de luxes acorde a la lectura de sensores, es decir, en caso de una anomalía como una sobrecarga la el sistema de iluminación se ajustará al mejor escenario.

Cumplimiento del Objetivo Específico 4

Y finalmente, se evaluó la funcionalidad y viabilidad de la aplicación del sistema, analizando las señales de control y mediciones obtenidas mediante la lectura de los sensores utilizados durante la realización de esta propuesta.

De acuerdo a lo indicado por [Al Balushi and Hussain \[2021\]](#), en su estudio basado en el sistema de ahorro de energía y monitoreo de la calidad del aire basado en IoT, el presente trabajo cumple con la sinergia del IoT y la red de sensores inalámbricos, ya que de este modo se puede gestionar la data recolectada de manera centralizada.

Asimismo, se destaca el trabajo de [Richard Chávez Vásquez \[2019\]](#), el cual en su proyecto de Análisis y desarrollo de un prototipo automatizado para el monitoreo de un invernadero a través de tecnologías de bajo costo, se secunda la moción de utilizar recursos y/o herramientas open sources para el desarrollo de este tipo de propuestas, debido a que gracias a la amplia documentación que brinda este tipo de tecnologías se hace más sencillo implementarlos y manipularlos de acuerdo a las necesidades que se presenten.

En cuanto a las ventajas y desventajas que se presentan en la elaboración de este proyecto se destacan:

Ventajas.-

- Al utilizar tecnologías open sources es la posibilidad que nos brinda de modificar el código fuente de los sistemas acorde a lo que se necesita, lo que facilita el trabajo de sobremanera, ya que se podrá adaptar diversas tecnologías de manera sinérgica.
- En la documentación de cada una de las librerías utilizadas está el código fuente de cada uno de los sensores utilizados.
- Al trabajar con IoT se brinda la facilidad de gestionar la data capturada en tiempo real y en cualquier dispositivo que tenga acceso a Internet.

Desventajas.-

- Se presentan ciertas limitaciones por la regularización de frecuencias para la tecnología LoRa en Ecuador.
- En Europa, se utiliza un plan de frecuencia sin licencia de 868 MHz, mientras que en Norteamérica es de 915 MHz y en Asia es de 433 Mhz,

por lo que en Ecuador estas frecuencias de espectro radioeléctrico están destinados para otros usos.

5.2. Recomendaciones

Sugerencias para futuros investigadores

- Comprobar qué placas se acoplan de mejor manera acorde a las tecnologías que se utilizarán.
- Verificar que las conexiones sean correctas para evitar vicisitudes como falsos contactos o error en el posicionamiento de pines.
- Guardar en un lugar seguro el Gateway ID.
- Verificar las frecuencias seleccionadas al momento de configurar la plataforma IoT.
- Al momento de diseñar el dashboard y personalizarlo, tratar de hacerlo lo más intuitivo posible para garantizar la experiencia del usuario.
- Al momento de codificar el código del Arduino, verificar en qué pines están declarados los equipos que se van a utilizar, ya sea en puertos digitales o analógicos.
- Verificar el funcionamiento de esta propuesta en plataformas como Ubidots y AWS IoT.

Bibliografía

- R. N. A. 006/18. «prestación del servicio de alumbrado público general»regulaciÓn, el directorio de la agencia de –arconel–, y control de electricidad, 2018. URL <https://www.regulacionelectrica.gob.ec/resoluciones-directorio/>.
- B. Abinaya. Iot based smart and adaptive lighting in street lights. pages 195–198, 2017.
- S. Ahmed. Iot based smart cities and cloud data storage. 68:72–78, 2020.
- I. Al Balushi and S. M. Hussain. Iot based air quality monitoring and controlling in underground mines, 2021.
- S. J. Alcolea. Soluciones tecnológicas para smart cities, 2019. URL <http://www.enertic.org/CentroConocimiento?param1=3137>.
- ALFAIOT. The internet of all the things, 2019. URL <https://alfaiot.com>.
- N. C. Almeida, R. P. Rolle, E. P. Godoy, P. Ferrari, and E. Sisinni. Proposal of a hybrid lora mesh / lorawan network. pages 702–707. Institute of Electrical and Electronics Engineers Inc., 6 2020. ISBN 9781728148922. doi: 10.1109/MetroInd4.0IoT48571.2020.9138206.
- M. P. M. Andrés. “implementaciÓn de una interfaz vga de bajo costo para la utilizaciÓn de monitores antiguos como paneles de informaciÓn”, 2019.
- R. Anitha, M. Nishitha, K. Akhila, K. S. Anusha, and G. Srilekha. Iot based smart and flexible lightning in streets. 7:291–294, 2018.
- C. M. Bruno. El origen del iot, 2017. URL <https://es.linkedin.com/pulse/el-origen-del-iot-bruno-cendón#:~:text=\T1\textquotedblleftSituviésemosordena>.

- S. H. Caballero. Estudio en detalle de lorawan. comparación con otras tecnologías lpwan considerando diferentes patrones de tráfico. 2020. URL <http://hdl.handle.net/10609/106369>.
- D. Carolina and C. Ortiz. Desarrollo de una red punto a punto con tecnología lora y servidor web, 2020.
- L. Chen, G. Xu, Q. Zhang, and X. Zhang. Learning deep representation of imbalanced scada data for fault detection of wind turbines. *Measurement: Journal of the International Measurement Confederation*, 139:370–379, 2019. ISSN 02632241. doi: 10.1016/j.measurement.2019.03.029. URL <https://doi.org/10.1016/j.measurement.2019.03.029>.
- Y. Chen, Y. Sun, N. Lu, and B. Wang. Journal of network and computer applications channel-reserved medium access control for edge computing based iot. *Journal of Network and Computer Applications*, 150:102500, 2020. doi: 10.1016/j.jnca.2019.102500. URL <https://doi.org/10.1016/j.jnca.2019.102500>.
- G. Corrente. Internet of things random motion nodes empowering opportunistic networks for smart cities. *Internet of Things*, 11:100258, 2020. ISSN 2542-6605. doi: 10.1016/j.iot.2020.100258. URL <https://doi.org/10.1016/j.iot.2020.100258>.
- D. G. Y. C. R. M. CÁRDENAS. Protocolo lora para implementación de redes iot en smart cities, 2018. URL http://profesores.elo.utfsm.cl/~agv//elo323.ipd438/2s18/projects/reports/RetamalCardenasGonzalez/Informe_Lora.pdf.
- M. Cárdenas, D. Gonzáles, and C. Retamal. Protocolo lora para implementación iot en smart cities. *Redes De Computadores*, pages 1–5, 2018.
- U. Electronics. Módulo sensor de voltaje 0 25v dc, 2022. URL <https://uelectronics.com/producto/modulo-sensor-de-voltaje-0-25v-dc/>.
- F.-J. Ferrandez-Pastor, L. Maciá Soler, M. Platero Horcajadas, D. Silveira Madrid, et al. Plataforma iot basada en paneles de monitorización (dashboard), 2021.
- P. P. Garcés. Redes de Área extensa para aplicaciones de iot: modelado de comunicaciones sigfox. 8 2017. URL <https://riunet.upv.es:443/handle/10251/86052>.

- I. George and R. Velazco. Autor: Samuel aguilar zavaleta para optar el título profesional de ingeniero de telecomunicaciones, 2020.
- S. Hernández. Estudio en detalle de lorawan. comparación con otras tecnologías lpwan considerando diferentes patrones de tráfico. 2020.
- M. Ismael. Sistema de administracion inteligente de alumbrado publico. 5 2018.
- C. Katherine and S. Daniela. Análisis de rendimiento de la tecnología lorawan aplicado al desarrollo de un sistema de monitoreo de calidad de aire en la unidad educativa santo tomás apostol riobamba, 2019.
- Lilypad. Mouser, 2019.
- A. Loubany, S. Lahoud, and R. El. Adaptive algorithm for spreading factor selection in lorawan networks with multiple gateways. *Computer Networks*, 182:107491, 2020. doi: 10.1016/j.comnet.2020.107491. URL <https://doi.org/10.1016/j.comnet.2020.107491>.
- T. Lucas. Conozca tagoio , la primera herramienta en la nube para el desarrollo de soluciones iot homologadas de khomp principales funciones de la plataforma. 2019. URL <https://www.khomp.com/es/tagoio-solucion-iot/>.
- S. Majdi. Practical evaluation of an effective intelligent central dimming strategy applied to public lighting network. pages 19–20, 2017.
- Naylmap. Tutorial sensor de corriente ac no invasivo sct-013, 2018. URL https://naylampmechatronics.com/blog/51_tutorial-sensor-de-corriente-ac-no-invasivo-sct-013.html.
- M. d. C. Olleros Guerrero. Desarrollo de un sistema basado en tecnologías iot y dlt para garantizar la calidad a lo largo de la cadena de suministro aplicado a transporte refrigerado, 2020.
- PCEL. Sensor de movimiento dsc lc-151 para exteriores, tecnología dual pir y doppler., 2018.
- A. Rafael and J. O. Jiménez. Mitigación de pérdida de paquetes en una red lorawan para aplicaciones sin línea de vista, 2019.
- D. M. B. Richard Chávez Vásquez. “análisis y desarrollo de prototipo automatizado para el monitoreo de un invernadero orgánico en la finca

- “la piedra” recinto de cerecita, en la localidad de bajada de chanduy, por medio de alternativas tecnológicas de bajo costo.”. 2019.
- A. E. Romo-González and M. de los Ángeles Villalobos-Alonzo. Aplicaciones y desarrollo de prototipos con internet de las cosas. *Revista de Tecnologías Computacionales*, pages 1–7, 2019. doi: 10.35429/joct.2019.9.3.1.7.
- A. L. Rosa. Lpwan como base de la comunicaiion iot, 2018.
- K. Rose, S. Eldridge, and L. Chapin. Octubre de 2015 para entender mejor los problemas y desafíos de un mundo más conectado, 2015.
- I. O. P. C. Series and M. Science. Designing smart greenhouse systems using scada based on iot. 2020. doi: 10.1088/1757-899X/850/1/012002.
- E. Store. Electro store, 2019.
- A. Systems. Sensor de movimiento para exteriores de doble tecnología • detección de microondas basada en el concepto doppler • compensación de temperatura • procesamiento de señal de microcontrolador • protección contra manipulaciones frontal y posterior • indicación, 2022. URL <https://www.alarmsystemsecuador.com.ec/producto/sensor-de-movimiento-para-exteriores-de-doble-tecnologia/>.
- TIBCO. ¿qué es el internet de las cosas (iot)? | tibco software, 2021. URL <https://www.tibco.com/es/reference-center/what-is-the-internet-of-things-iot>.
- J. B. Viñas. Evaluación de tecnologías lpwan para escenarios de smart cities. pages 1 – 36, 2018. URL <http://repositorio.upct.es/handle/10317/7296>.
- W. Waluyo, A. Widura, F. Hadiatna, and D. Anugerah. Comparative power and energy consumptions between scheduled and fuzzy controlling on an iot-based vertical farming. pages 278–282. Institute of Electrical and Electronics Engineers Inc., 2021. ISBN 9781665443548. doi: 10.1109/ICHVEPS53178.2021.9601038.
- S. A. Zavaleta. Diseño de una solución basada en iot empleando lorawan para el monitoreo de cultivos agrícolas en Perú, 2020.