



PONTIFICIA
UNIVERSIDAD
CATÓLICA DE
VALPARAÍSO



Fernanda Javiera Coronado Toledo

Estudio y Desarrollo de un Sistema de Monitoreo de Radiofrecuencia Multivariable con Integración de Alarmas Preventivas

Informe Proyecto de Título de Ingeniero Civil Electrónico



**Escuela de Ingeniería Eléctrica
Facultad de Ingeniería**

Valparaíso, 26 de septiembre de 2018



Estudio y Desarrollo de un Sistema de Monitoreo de Radiofrecuencia Multivariable con Integración de Alarmas Preventivas

Fernanda Javiera Coronado Toledo

Informe Final para optar al título de Ingeniero Civil Electrónico,

aprobada por la comisión de la

Escuela de Ingeniería Eléctrica de la

Facultad de Ingeniería de la

Pontificia Universidad Católica de Valparaíso

conformada por

Sr. Francisco Alonso Villalobos

Profesor Guía

Sr. Felipe Leighton González

Segundo Revisor

Sr. Sebastián Fingerhuth Massman

Secretario Académico

Valparaíso, 26 de septiembre de 2018

A mi madre, Sandra Toledo.

Agradecimientos

En primer lugar, le doy gracias a mi madre, Sandra Toledo, por el amor y apoyo incondicional que siempre me ha brindado, sin ella no hubiese llegado hasta acá. A mi hermana, Almendra Coronado, que a pesar de su corta edad me ha dado grandes enseñanzas de vida. Agradezco además a mi familia por siempre creer en mí y apoyarme cuando más lo necesité.

También agradezco a mis amigos de la Universidad que me han acompañado estos años de estudio, sufriendo por las pruebas y compartiendo grandes momentos.

Sin ustedes, esto no sería posible.

Valparaíso, 26 de septiembre de 2018

F.C.

Resumen

El monitoreo de distintos factores a tiempo real es muy importante en muchas áreas, una de éstas es en Agricultura, específicamente en invernaderos, donde se realizan monitoreos de distintas variables para verificar que se tienen los valores adecuados para la producción, por lo que se realiza un estudio de algunos cultivos para ver cuáles son todas las variables importantes. El monitoreo se puede llevar a cabo de distintas maneras, además se puede incorporar un sistema de alarma, la cual tiene distintas formas de activarse, por lo que se ve cuál es el desarrollo que tienen en Chile y el mundo. Estos estudios permitieron determinar la forma de realizar el sistema de monitoreo y alarma, llegando a un sistema de monitoreo inalámbrico mediante IP y una alarma vía E-mail.

Una vez determinado el sistema a realizar, se hicieron las configuraciones necesarias de los distintos dispositivos a utilizar, tales como, Arduino, Raspberry Pi, módulo XBee y sensores, y se determinan dos nodos, el Nodo “Dispositivo Final” y el Nodo “Coordinador”. Luego se realizó el diseño final para obtener información representativa del lugar, para esto se debieron realizar pruebas en terreno, de forma de determinar la cantidad necesaria de Nodos “Dispositivo Final” al interior del invernadero.

Se realizó la implementación del sistema de monitoreo y alarma, donde se creó un hardware óptimo para su utilización, dejando todas las conexiones de los dispositivos al interior de una carcasa diseñada en 3D. Además, se realizaron las programaciones en los distintos lenguajes para lograr la comunicación entre dispositivos, la creación de la base de datos y finalmente la interfaz gráfica que permite la visualización de los datos mediante gráficos. También se realizó la programación necesaria para la creación de la alarma vía E-mail.

Para validar este proyecto, fue necesario realizar algunas pruebas en terreno. Para esto, se realizaron pruebas durante un periodo de aproximadamente 5 horas en un invernadero de Limache y otro de Quillota, comprobando que el sistema funciona correctamente y es bastante efectivo debido a que se puede visualizar en tiempo real.

Finalmente, se realizó un estudio económico, donde se obtuvo el valor final del proyecto y se hace un estudio de rentabilidad, comprobando que este sistema es económicamente viable.

Palabras claves: monitoreo, invernadero, alarma, inalámbrico, XBee, Base de Datos, interfaz

Abstract

Monitoring different factors in real time is very important in several areas, one of these are the Agriculture, specifically the greenhouses, where variables are monitoring to verify the appropriate values for production. A study of some crops is carried out to see which are the important variables. The monitoring can be done in different ways, also it can incorporate an alarm system, which has different methods to activate so it is possible to observe the development in Chile and also world. These studies allow to determine the way to perform the monitoring and alarm system, reaching at a wireless monitoring system via IP and an alarm via E-mail.

Once determined the system to realize, it is require configurations of different devices, such as Arduino, Raspberry Pi, XBee module and sensors, and two nodes were determine, the Node "Final Device" and the Node "Coordinator" " Then the final design is made to obtain representative information of place, for this is necessary do tests in the field, in order to determine the essential quantity of "Final Device" nodes inside the greenhouse.

It was done the implementation of monitoring and alarm system, where a comfortable hardware was created for using, thus all device connections are inside a housing designed in 3D. In addition, some programmings were done in different languages to have communication between devices. The creation of the database and finally the graphical interface that allows visualization of data through graphics. There was an alarm creation of programming for the via E-mail.

To validate this project, it was necessary to perform some tests directly in the greenhouses. For this, tests were made during a period of approximately 5 hours in a Limache greenhouse and another one in Quillota. It was verify that the system worked correctly and is quite effective because it can be viewed in real time.

Finally, an economic study was done, where it was obtaining a final value of the project and a profitability study is made, to confirm that this system is economically viable.

Keywords: Monitoring, greenhouse, alarm, wireless, XBee, Database, interface

Índice general

Introducción.....	1
Objetivo General	3
Objetivos Específicos	3
1 Antecedentes del Proyecto.....	5
1.1 Problemática	5
1.2 Solución propuesta.....	6
1.3 Variables a monitorear	7
1.3.1 Tomate	7
1.3.2 Limón	8
1.3.3 Palta	8
1.3.4 Pimiento	8
1.3.5 Elección de variables	8
2 Estado del Arte	10
2.1 Sistemas de monitoreo de temperatura y humedad	10
2.1.1 Monitoreo en el lugar.....	10
2.1.2 Monitoreo por GSM o por e-mail.....	11
2.1.3 Monitoreo por IP	12
2.1.4 Monitoreo a través de la nube (IoT)	12
2.1.5 Comparación Sistemas de Monitoreo	13
2.2 Sistemas de Alarma de temperatura y humedad	13
2.2.1 Alarma por medio de sirena y/o luz.....	14
2.2.2 Alarma por medio de E-mail, SMS o WhatsApp	14
2.2.3 Comparación Sistemas de Alarma	15
2.3 Empresas de Sistemas de Monitoreo y Alarma en Chile y el Mundo	16
2.3.1 Meldic, Chile	16
2.3.2 Unisource, Chile	16
2.3.3 Epsilon, Chile	17
2.3.4 Banner Engineering, EEUU	17
2.3.5 Etherpower, EEUU	17
2.3.6 Novus, Brasil	18

2.3.7 Resumen Empresas	18
2.4 Proyecto de medición de temperatura en sala de la EIE	18
3 Marco Teórico	20
3.1 Descripción de Hardware	20
3.1.1 XBee	20
3.1.2 Raspberry Pi	25
3.1.3 Arduino	27
3.1.4 Sensores	28
3.1.5 DHT 22	28
3.1.6 BH-1750	29
3.1.7 MG-811	29
3.2 Descripción de Software	30
3.2.1 X-CTU	30
3.2.2 Arduino IDE	31
3.2.3 Python	32
3.2.4 MySQL	32
3.2.5 phpMyAdmin	33
3.2.6 PHP	33
3.2.7 Apache	34
4 Implementación	35
4.1 Invernadero	35
4.2 Previo a la realización del proyecto	35
4.2.1 Configuración XBee	36
4.2.2 SSH y VNC	38
4.3 Diseño Final	39
4.3.1 Dispositivo Final	39
4.3.2 Coordinador	41
4.3.3 Diseño	42
4.3.4 Diseño de Hardware	43
4.4 Realización Monitoreo	45
4.4.1 Comunicación entre dispositivos	46
4.4.2 Base de Datos	48
4.4.3 Interfaz gráfica	52
4.5 Realización Alarma	54
4.5.1 Envío por E-mail	54
4.5.2 Límites	55
4.6 Diseño Carcasa 3D	56
4.7 Pasos a seguir para instalación	57
5 Resultados	59
5.1 Pruebas sensores	59
5.2 Invernadero Limache	59

5.3 Invernadero Quillota	63
5.4 Incorporación alarma.....	68
6 Evaluación Económica.....	70
6.1 Componentes	70
6.2 Mercado de proveedores.....	71
6.3 Análisis de costos	72
6.4 Estudio de Rentabilidad	73
6.5 Comparación con mercado actual	74
Discusión y conclusiones.....	76
Bibliografía	80

Introducción

El monitoreo de diversos factores y variables es de gran importancia hoy en día en distintas áreas como la medicina, transporte de alimentos perecederos, la agricultura, entre otros. En este último, la tecnología ha tomado fuerza y cada vez se utiliza más para mejorar o automatizar procesos, por lo que la implementación de sistemas de monitoreo, alarma y control en invernaderos se hace cada vez más común.

En agricultura se utiliza cada vez más la producción de cultivos bajo invernaderos, debido a que se logran condiciones climáticas que al aire libre sería imposible conseguir. Estos permiten tener un mayor control del entorno donde crecen los cultivos, mejorando la producción y calidad del producto, por lo que es de gran importancia tener conocimiento de los valores que tienen las variables que son influyentes para el cultivo, estos valores generalmente son medidos por personal entendido en el tema del cultivo bajo invernadero, de forma que pueda realizar y analizar las mediciones de las distintas variables. Sin embargo, el hombre comete muchos errores en la toma de datos y el análisis de las mediciones significa mucho tiempo, lo cual muchas veces no permite detectar a tiempo los riesgos de los cultivos que más tarde afectará directamente en su producción.

Hoy en día, existen algunas empresas que ofrecen sistemas que permiten realizar el monitoreo de forma automática, donde la toma de datos de las distintas variables se obtiene automáticamente a través de sensores, los cuales tienen un margen de error muy pequeño y permite tener valores más precisos que las mediciones realizadas por el personal especializado. Además, al sistema de monitoreo que ofrecen estas empresas, se le puede incorporar una alarma y/o control de variables, donde ambos consisten en reaccionar cuando la medición de una variable se encuentra fuera de los límites establecidos previamente. El sistema de alarma es un sistema pasivo, no evita una situación, pero al activarse permite realizar alguna acción al interior del invernadero, mientras que el sistema de control realiza tal acción de forma automática, el problema de este último es que es muy costoso debido a que se debe adaptar el invernadero para la automatización de las acciones.

La tecnología avanza a pasos agigantados, lo cual ha permitido el desarrollo de los sistemas de monitoreo a lo largo del tiempo, permitiendo que estos sistemas sean mucho más fáciles y rápidos de manejar. Una de las tecnologías que va tomando cada vez más fuerza es el internet de las cosas (IoT), la cual ha permitido que objetos y dispositivos interactúen con entornos virtuales de datos, permitiendo la conexión de lo físico con el internet, lo cual facilita notablemente el monitoreo debido a que los datos obtenidos automáticamente del invernadero se podrán visualizar en cualquier parte del mundo que se tenga conexión a internet.

Generalmente los invernaderos se encuentran insertados en un medio agrícola, alejados de las ciudades, lo cual dificulta la implementación de algún sistema electrónico, ya que se debe considerar que se tendrán grandes distancias entre el invernadero y alguna casa u oficina donde se pueda realizar el monitoreo. Al tener estas distancias se hace complicado realizar las conexiones de dispositivos con cables, por lo que se debe considerar algún medio inalámbrico.

Los módulos XBee son dispositivos capaces de comunicarse de forma inalámbrica unos con otros. Esta comunicación puede ser muy simple como conectar dos dispositivos, o bien, puede ser muy compleja y conectar muchos dispositivos. Son una solución inalámbrica fácil de utilizar, teniendo distintas opciones de comunicación dependiendo la dificultad de la red, donde en casos de redes amplias conviene un envío de información mediante tramas, obteniendo la información de forma ordenada, dándole mayor facilidad a una gran red. Además, al incorporar un microcontrolador a la red, se pueden utilizar en infinidad de aplicaciones.

Raspberry Pi es un microcontrolador y mini ordenador que permite realizar múltiples funciones dependiendo las necesidades que se requieran. Al ser un microcontrolador permite la conexión de diversos componentes como los módulos XBee, y como mini ordenador le permite funcionar como servidor web, es decir, almacenar y manejar archivos o datos y enviarlas a través de la red, todo esto del lado del servidor y realiza la conexión con el cliente generando una respuesta. De esta forma, la Raspberry Pi almacenará y organizará datos para realizar el monitoreo de las variables y luego se podrá visualizar desde cualquier dispositivo conectado a la misma red. Además, al almacenar los datos se puede trabajar con ellos, de forma de realizar un sistema de alarma.

La evolución que ha tenido la electrónica y la informática ha mostrado que estas dos ciencias tienen una relación muy estrecha, ya que para cualquier acción que desee realizar algún dispositivo electrónico se debe realizar la programación necesaria para que tal acción se cumpla. La programación es el lenguaje que permite la comunicación entre dispositivos electrónicos y entre dispositivo y el hombre, como lo es el celular, la televisión o la lavadora. Esta se hace necesaria en la electrónica para obtener datos de distintos sensores, dar una orden a algún otro dispositivo o para activar un actuador, lo cual permite que la electrónica sea muy variable y se puedan realizar una infinidad de proyectos, ya que el límite lo pone quien realiza el proyecto. La placa Raspberry Pi debe ser programada para que cumpla las distintas funciones requeridas para lograr el monitoreo de variables, tales como, recibo de datos provenientes del invernadero, servidor web o realización de alarma.

Con la idea de hacer más simple e intuitivo el uso de los computadores, nació la llamada interfaz gráfica de usuario (IGU), la cual permite interactuar con el sistema más fácilmente, ya que antes las interfaces de usuario no eran gráficas y solo consistían en comandos que se debían recordar. Una interfaz es el medio que permite establecer la comunicación entre dos sistemas que no hablan el mismo lenguaje y también es la cara visible de los programas tal y como se presenta a los usuarios para que interactúen con el computador. La interfaz requiere una pantalla, donde se presentan las opciones que el usuario puede elegir dentro del sistema mediante una serie de menús e iconos.

Un diseño 3D es un conjunto de técnicas que permiten proyectar un objeto digital en tres dimensiones mediante la generación de gráficos tridimensionales. Esta herramienta permite tener una visión diferente de los objetos a imprimir, ya que se pueden plasmar concretamente las ideas que se tienen para la confección de estos, y de esta forma verificar si lo que se está imaginando es realmente viable o si se debe realizar algún cambio.

Este proyecto utiliza distintos dispositivos, tales como, Raspberry Pi, módulos XBee, entre otros, los cuales en conjunto permitirán la realización a nivel de hardware de un sistema de monitoreo y alarma. Además, a nivel de software, se utilizarán distintos lenguajes de programación para lograr las distintas conexiones requeridas, una interfaz gráfica para poder visualizar los datos de forma fácil y cómoda y el diseño 3D para poder proteger los dispositivos.

Objetivo General

El objetivo general consiste en desarrollar un sistema de monitoreo y de alarma para la aplicación en invernaderos utilizando placas de desarrollo XBee, incorporando un sistema de alarmas preventivas, para la medición de temperatura, humedad, luminosidad y dióxido de carbono.

Objetivos Específicos

Se debe cumplir una serie de objetivos específicos, de esta forma se podrá llegar a cumplir completamente el objetivo general. Estos objetivos permitirán la realización de este proyecto, realizando la investigación necesaria para que finalmente se realice la implementación del proyecto. Los objetivos específicos de este proyecto son:

- Revisión del estado del arte acerca de los sistemas de monitoreo de temperatura y humedad que existen hoy en día.
- Revisión del estado del arte acerca de los sistemas de alarma existentes.
- Estudiar las variables más relevantes e influyentes a monitorear en los cultivos bajo invernadero.
- Estudiar la placa de desarrollo XBee.
- Estudiar la placa Raspberry Pi para el sistema de monitoreo y alarma.
- Estudiar los lenguajes de programación para determinar cuáles y cómo se utilizarán para el desarrollo del sistema de monitoreo y alarma.
- Desarrollar un prototipo para la medición de las variables de manera inalámbrica a través de las placas XBee.

- Desarrollar un prototipo de sistema de monitoreo con incorporación de alarmas
- Validar prototipo creado de sistema de monitoreo y sistema de alarmas.

1 Antecedentes del Proyecto

1.1 Problemática

Hoy en día, es necesario realizar monitoreo en distintas áreas, como lo son la medicina, el transporte de alimentos perecederos o la agricultura. En esta última, la utilización de invernaderos es muy común debido a que se pueden obtener las condiciones necesarias para poder producir adecuadamente. En Chile existe una gran cantidad de invernaderos, tanto industriales, que son de gran tamaño, como particulares, los cuales son más reducidos.

Los invernaderos son estructuras de diversas formas y tamaños dependiendo del requerimiento y están cubiertas por plástico o vidrio para así poder retener de mejor forma el calor. Estas estructuras tienen la capacidad de generar condiciones climáticas ideales para el cultivo de hortalizas cuando las condiciones al aire libre no son las óptimas para que se produzcan de buena manera [1]. Cada tipo de hortaliza requiere condiciones climáticas específicas para su cultivo y si estas no se cumplen, no se obtendrá un buen resultado. Para tener un buen cultivo, es importante la precisión de los factores que afectan en este, sin embargo, el hombre comete muchos errores en la toma de datos y no se logra la precisión que se necesita, lo que podría ocasionar que las hortalizas no obtengan las características adecuadas. Es debido a esto que la automatización de invernaderos es una buena opción para mejorar la eficiencia de estos, donde las mediciones son realizadas a través de sensores para obtener un valor mucho más preciso.

Las grandes industrias de la cosecha de frutas y verduras utilizan cada vez más los invernaderos para poder producir en épocas donde, en condiciones naturales, es imposible producir. Estos invernaderos cuentan con alta tecnología ya que, al ser de gran tamaño, requieren monitoreo constante, el cual no es preciso ni económicamente viable al ser realizado por personas. Otro factor muy importante es el control de las variables que se están monitoreando, ya que, al requerir condiciones específicas, si hay alguna variable que sale de los límites óptimos, esta se debe controlar para que vuelva a su valor ideal. Las tecnologías mencionadas anteriormente tienen un precio muy elevado debido a que generalmente se utilizan equipos costosos y sofisticados y la mayoría funciona con tecnología más avanzada.

Por otra parte, los invernaderos particulares son mucho más pequeños que las grandes industrias, por lo que se hace mucho más difícil tener grandes tecnologías debido al alto costo que estas traen consigo. Generalmente, los particulares no poseen formas de medir las distintas variables y sólo trabajan con la experiencia que les ha dado la vida en lugares agrícolas. Sin embargo, existen algunos particulares que trabajan con personal dedicado a verificar que las condiciones sean las adecuadas, pero aun así las mediciones de las personas no son precisas y la producción no será la adecuada.

El desarrollo de un sistema de control implicaría realizar grandes modificaciones en algún invernadero, debido a que se deben instalar todo el equipamiento necesario para lograr el control de forma automática. Las grandes empresas poseen los recursos para tener una inversión inicial capaz de confeccionar invernaderos con los equipos de control, sin embargo, las pequeñas empresas y particulares en general no poseen los recursos y habría que realizar todas las modificaciones, es por esto que no se puede realizar este sistema. Sin embargo, se puede pensar en la realización de un sistema de alarma, la cual generará la alerta cuando alguna variable se salga de sus límites óptimos.

Debido a lo comentado anteriormente, se propone el desarrollo de un sistema de monitoreo de algunos factores importantes en un invernadero, como lo son la temperatura y humedad, incorporando alarmas preventivas, a un costo menor y al alcance para que pueda ser utilizado por pequeñas empresas y particulares.

1.2 Solución propuesta

Los invernaderos se encuentran generalmente alejados de alguna casa u oficina, por lo que para la realización de un sistema de monitoreo y alarma se piensa en un sistema inalámbrico, de esta forma se pueden tener los componentes de medición en el invernadero mientras que los componentes encargados de procesar y gestionar los datos se encuentren en alguna oficina a distancia.

Para lograr la realización del sistema de monitoreo y alarma se requiere de un microcontrolador que procese la información de las variables a medir para que luego la muestre o trabaje con ella, para esto se utilizará una placa Raspberry Pi. Además, se requieren componentes para realizar las mediciones de las variables al interior del invernadero, por lo que utilizarán distintos sensores. Se tendrán dispositivos a interior del invernadero y otros en una casa u oficina a distancia. Para realizar las mediciones directamente en el invernadero se utilizarán los sensores de las distintas variables a medir, un Arduino para recibir y, en caso de ser necesario, transformar la información de los sensores a valores representativos, y un módulo XBee, que es una antena de radiofrecuencia la cual funcionará como transmisor y se encargará de realizar el envío inalámbrico a la oficina a distancia. Por el otro lado, en la oficina, los componentes que se utilizarán son el módulo XBee, que funcionará como receptor y recibirá la información desde el módulo XBee transmisor ubicado en el invernadero, y una placa Raspberry Pi, la cual se encargará de procesar la información recibida para lograr el monitoreo y, además, trabajar con la información para lograr la alarma. En la Figura 1-1 se muestra el diagrama de conexiones necesarias para lograr la realización del sistema de monitoreo y alarma.

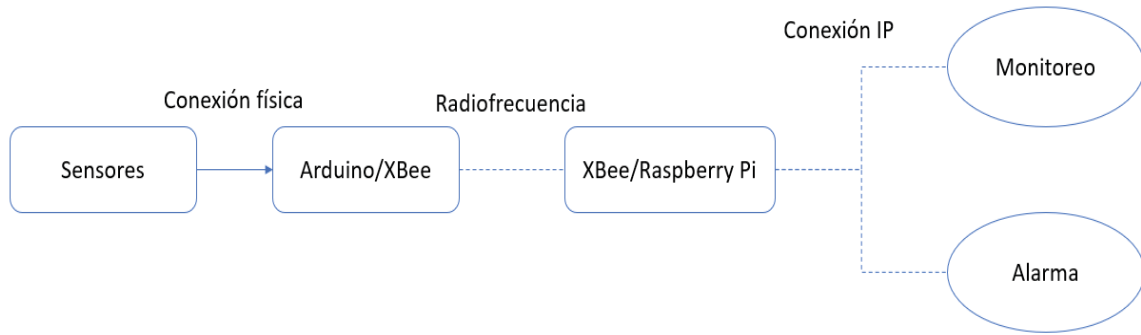


Figura 1-1 Diagrama Conexiones

1.3 Variables a monitorear

En cultivos bajo invernadero se tienen condiciones climáticas ideales para que crezcan de la mejor manera posible, debido a que se pueden conservar distintas variables en los límites óptimos para el crecimiento y de esta forma poder producir en épocas o lugares donde sería imposible producir al aire libre. Para determinar las variables importantes en cultivos bajo invernadero se analizarán cuatro cultivos distintos, que son algunos de los que más se producen en la región de Valparaíso, tomate, limón, palta y pimiento. Esto con el fin de ver las variables que se van a monitorear, de forma que el monitoreo sea lo más útil posible para los clientes.

1.3.1 Tomate

En el cultivo de tomates es muy importante el clima y hay algunas variables que influyen directamente para lograr un buen producto. El tomate prospera mejor en climas secos con temperaturas moderadas, donde su temperatura óptima varía entre 18 y 30°C, tener temperaturas sobre los 30° pueden ocasionar diversos trastornos, ya sea en la maduración, precocidad o color [2]. Otro factor importante es la humedad relativa, donde sus valores óptimos varían entre 60% y 80%, humedades relativas mayores favorecen el desarrollo de enfermedades fungosas y bacterianas y dificultan la fecundación, ya que el polen se compacta abortando parte de las flores [2], además, la combinación de altas temperaturas con baja humedad puede generar aborto floral y baja viabilidad del polen. Otro factor importante es el Dióxido de Carbono (CO₂), ya que es esencial para que se produzca la fotosíntesis, los tomates crecen mucho mejor en invernaderos donde la concentración es entre 1000 y 2000 ppm [3]. Por otra parte, está la luminosidad la cual cumple un rol importante debido a que el tomate requiere al menos 6 horas diarias de luz para florecer, en valores reducidos pueden incidir de forma negativa sobre este proceso y la fecundación [2].

1.3.2 Limón

El cultivo de limón crece de mejor forma en regiones tropicales y subtropicales y tiene algunos factores importantes. La temperatura es un factor limitante del cultivo, las medias favorables oscilan entre 10 y 24°C, al tener temperaturas mayores de 35°C pueden ocasionar trastornos vegetativos y aceleran la maduración de los frutos y con temperaturas menores de 12°C afectan el crecimiento vegetativo. La humedad relativa también influye en la calidad de la fruta, el rango óptimo es entre 40 y 70 %, cuando es muy alta favorece el desarrollo de enfermedades causadas por hongos. Otro factor importante a considerar es la luz solar debido a que se necesita una alta luminosidad [4]. El CO₂ también es importante, ya que así permite la fotosíntesis y no suele sobrepasar el 0.5 % para evitar que la respiración del fruto tome sabores extraños [5]

1.3.3 Palta

El cultivo de palta crece de mejor forma en climas subtropicales. La temperatura es muy importante, ya que es muy sensible a las bajas temperaturas y sufre grandes daños, los valores favorables son entre 20 a 25°C en el día y 10°C en la noche, así se tendrá una exitosa fecundación y una buena cuaja. Con una humedad relativa baja se puede agravar una condición de estrés y con una alta humedad relativa se puede aliviar, el rango medio favorable es entre 62 a 85% de humedad relativa. Otro factor importante es la humedad del suelo, ya que, con un exceso de humedad en el suelo, agota el O₂ y aumenta el CO₂. Con suelos bien drenados y con 15% de oxígeno y 0.03% de dióxido de carbono se tiene un crecimiento adecuado [6]. La captura de luz por parte de la planta es de gran importancia debido a que influye en el tamaño de la fruta.

1.3.4 Pimiento

El cultivo de pimiento crece en climas muy calurosos, y soporta temperaturas de hasta 35°C, sin embargo, los límites óptimos son entre 18 y 28 °C. Con temperaturas menores a 18°C se presentan múltiples frutos de menor tamaño y frutos deformes, y con temperaturas sobre 32°C se tiene una deficiencia en la fecundación y se produce la caída de las flores [7]. Junto con la temperatura, otro factor de gran importancia es la luminosidad, la cual es imprescindible para la función clorofílica, floración y maduración del cultivo, este requiere como mínimo 3000 lux. Otro factor que influye en el cultivo es la humedad relativa, la cual tiene sus límites óptimos entre 50 y 70%, humedades sobre el límite favorece el desarrollo de enfermedades y dificultan la fecundación, y humedades bajo el límite óptimo ocasiona la caída de flores y frutos [7]. Además, es importante el dióxido de carbono para la fotosíntesis del cultivo, se requiere entre 200 y 500 ppm [8].

1.3.5 Elección de variables

Del estudio de los cuatro cultivos mencionados anteriormente, se puede observar que hay cuatro variables muy importantes en todos los cultivos: temperatura, humedad relativa, dióxido de carbono y luminosidad. Es por esto que es importante saber qué está pasando con estas variables al interior del invernadero, por lo que se realizará el monitoreo de estas cuatro variables, así se podrá ver los valores que toman y el comportamiento a lo largo del tiempo. En la Tabla 1-1 se muestran los rangos óptimos de cada variable elegida en los distintos cultivos estudiados.

Tabla 1-1 Rango de variables en cultivos

	Temperatura [°C]	Humedad [%]	Luminosidad [Lux]	Dióxido de carbono [ppm]
Tomate	18-30	60-80	2.000-15.000	1.000-2.000
Limón	10-24	40-70	1.000-15.000	200-500
Palta	15-25	60-70	200-15.000	200-800
Pimiento	18-28	50-70	3.000-20.000	200-500

Estos rangos son de gran importancia al momento de implementar la alarma, ya que se deberán determinar los límites máximo y mínimo de cada variable en los invernaderos para que la alarma se active.

2 Estado del Arte

2.1 Sistemas de monitoreo de temperatura y humedad

Un Sistema de monitoreo permite supervisar alguna situación a través de la visualización en una pantalla de información de entrada, generalmente proveniente de sensores. Una vez visualizada la información, se pueden sacar conclusiones, por ejemplo, se visualizan en un gráfico los datos de temperatura de una habitación, lo cual permitirá ver a qué hora fue la temperatura máxima y mínima o qué sucede al ingresar más personas. Este último tiempo, la tecnología ha avanzado a pasos agigantados, lo cual ha permitido que el monitoreo de variables sea mucho más fácil para el cliente, ya que ahora solo basta con ver el teléfono o el computador para tener la información a tiempo real. Actualmente, existen diversas formas de monitorear factores, algunos de ellos son los que se indicarán a continuación.

2.1.1 Monitoreo en el lugar

El monitoreo en el lugar hace referencia al monitoreo que se hace en el mismo lugar donde se está haciendo la toma de datos, este no requiere de gran tecnología para su realización, ya que solo está compuesto por los componentes que están haciendo las mediciones, como sensores y microcontroladores. Generalmente, este monitoreo se hace a través de una pantalla led donde se obtiene la recepción de datos, donde la pantalla está inserta en el mismo lugar donde están todos los componentes de mediciones. Un ejemplo es “Invernador-5000” (Figura 2-1), el cual es un equipo diseñado para el control, monitoreo y registro de variables ambientales y muestra en pantalla la información de los sensores [9].



Figura 2-1 Invernador-5000. (fuente: www.solsis.cl)

2.1.2 Monitoreo por GSM o por e-mail

El monitoreo por GSM es donde se tiene un dispositivo que hace la toma de datos, los cuales se pueden enviar por mensaje de texto al teléfono que está sincronizado a tiempo real. Los datos se pueden solicitar o configurar el dispositivo para que envíe la información cada cierto tiempo. Un ejemplo son los dispositivos “Control GSM 5010 5011” (Figura 2-2), este es un equipo que permite monitorear y controlar equipos vía mensajes SMS, el cual está permanentemente conectada a un teléfono celular GSM. Puede enviar mensajes SMS con mensajes de alarma o con el valor de la variable [10].



Figura 2-2 Control GSM 5010 5011 (fuente: www.jmi.com.mx)

También está el monitoreo por e-mail el cual cumple la misma función que el monitoreo por GSM, la única diferencia es que la información se envía a un correo electrónico. Un ejemplo de este monitoreo es el dispositivo “SureCross FlexPower” (Figura 2-3), el cual tiene una batería interna conectado directamente a un sensor de temperatura y humedad. Este dispositivo puede programar eventos específicos, registrar y guardar los datos y enviar mensajes de estado por e-mail, además, cuando las mediciones se encuentran fuera del rango aceptable, se envía una alerta por e-mail.



Figura 2-3 SureCross FlexPower (fuente: www.bannerengineering.com)

2.1.3 Monitoreo por IP

El monitoreo por IP consiste en obtener la información desde la red de área local, es decir, se podrá ver la información obtenida desde cualquier dispositivo que se conecte a la misma red de internet. Ya no es necesario estar en el lugar para poder monitorear, si bien es cierto que en el monitoreo por GSM y por e-mail ya no era necesario estar en el lugar, con el monitoreo por IP es mucho más fácil, ya que se puede ver desde cualquier dispositivo, ya sea computador o celular conectado a la misma red que el dispositivo que entrega la información. Hoy en día existen varias empresas que ofrecen este tipo de monitoreo, un ejemplo es “EtherPower”, el cual ofrece una variedad de servicios entre los cuales está el monitoreo por IP. El equipo “SUN-32” (Figura 2-4) es uno de los equipos de monitoreo, el cual está diseñado para el monitoreo remoto, además puede enviar un e-mail cuando alguna variable excede algún rango [11].



Figura 2-4 SUN-32 (fuente: www.etherpower.net)

2.1.4 Monitoreo a través de la nube (IoT)

Una de las tecnologías más utilizadas en el último tiempo es el internet de las cosas (IoT), IoT es una arquitectura emergente basada en Internet global que facilita el intercambio de bienes y servicios entre redes de la cadena de suministro [12]. Por lo tanto, el monitoreo a través de IoT hace referencia a que los datos obtenidos por algún dispositivo serán subidos a la nube, lo cual permitirá ver la información desde cualquier lugar del mundo y desde cualquier dispositivo. Un ejemplo de este monitoreo es “Saveris 2” (Figura 2-5), el cual es un sistema compuesto de registradores de datos Wifi y un acceso a la nube Testo, los registradores hacen la toma de datos y luego los transmiten a la nube Testo. Además, si sobrepasa los límites de envía un e-mail o mensaje SMS.



Figura 2-5 Saveris 2 (fuente: www.testo.com)

2.1.5 Comparación Sistemas de Monitoreo

Existen diversas formas de realizar el monitoreo de temperatura y humedad y, dependiendo lo que requiere el cliente, se podrá elegir entre ellos. A continuación, se presenta una tabla comparativa de las distintas tecnologías utilizadas en sistemas de monitoreo.

Tabla 2-1 Comparación sistemas de monitoreo

	Monitoreo en el lugar	Monitoreo por GSM o E-mail	Monitoreo por IP	Monitoreo a través de la nube
Alcance (metros)	0	100000-1000000	10-10000	indeterminado
Número de dispositivos	-	1-10	15-50	indeterminado
Tipo de tecnología	-	WWAN	WLAN	WWAN
Costo (alto, medio, bajo)	Bajo	Medio	Alto	Alto

En la Tabla 2-1 se puede observar que, si bien el monitoreo por IP y a través de la nube son más costosos que los otros dos, son opciones mucho más interesantes debido a que permite visualizar la información a diversos dispositivos y, en el caso del monitoreo a través de la nube, se puede visualizar en cualquier parte del mundo.

2.2 Sistemas de Alarma de temperatura y humedad

Un sistema de alarma es un sistema pasivo, ya que no evita una situación anormal, pero sí permite advertir al cliente de ella [13]. Esto permite la reducción del tiempo de ejecución de alguna acción a tomar en función del problema que se presenta, lo cual permite evitar lamentaciones futuras. Este sistema puede variar dependiendo del lugar donde se requiera y qué se requiere alarmar, por ejemplo, en una casa, se puede instalar un sistema de alarma con una cámara y que avise cuando ingrese alguien, y si es un sistema más sofisticado podría tener incluso reconocimiento de personas y alertar cuando alguien extraño ingrese, o bien, una alarma de detección de incendios, la cual avisará cuando capte el humo y/o altas temperaturas. La alarma puede ser activada manual o automáticamente, este último requiere de sensores que capten las señales del entorno y dependerán de lo que se quiere alarmar, y se debe determinar algún límite para que se active la alarma.

Un Sistema de Alarma de temperatura y humedad tiene una alarma automática a través de un sensor de temperatura y humedad, el cual permitirá alertar cuando alguna de las variables es más baja o más alta de lo que se desea. Una vez activada la alarma, existen diversos medios de envío, los cuales pueden ser directamente en el lugar donde se está midiendo o enviar una alarma a distancia. A continuación, se indicarán algunos medios de envío de alarma.

2.2.1 Alarma por medio de sirena y/o luz

La alarma por medio de sirena y/o luz es una alarma que, al activarse, se enciende la sirena y/o luz en el lugar donde se tiene el Sistema de Alarma, de esta forma, quienes se encuentren en el lugar podrán realizar alguna acción dependiendo la alarma. Esta alarma no es completamente útil debido a que, si no hay alguna persona en el lugar, no se podrá tomar alguna decisión respecto a ella. Por ejemplo, se tiene una alarma de incendio en una habitación, la cual se activa al detectar 60 °C, el día que esta se active comenzará a sonar una sirena, pero si en las cercanías de la habitación no hay alguien, nadie la escuchará y el incendio continuará debido a que nadie realiza alguna acción (Figura 2-6).



Figura 2-6 Detector DTS (fuente: www.adt.cl)

2.2.2 Alarma por medio de E-mail, SMS o WhatsApp

El envío de la alarma por medio de E-mail, SMS o WhatsApp son envíos a larga distancia, y estos tres se diferencian en el modo de envío de la alarma, por lo tanto, al activarse la alarma, será enviada a un correo electrónico determinado en el caso de E-mail o a un número de teléfono previamente determinado en el caso de SMS y WhatsApp, sin necesidad de encontrarse en el lugar. Cada una tiene distintos límites de envíos: la limitación del envío por E-mail es que depende de internet, tanto por parte del Sistema de Alarma como del cliente al que se le envía el correo electrónico; la limitación del envío por SMS es que la tarjeta SIM utilizada en la alarma debe tener dinero para realizar el envío de SMS; la limitación del envío por WhatsApp es que el sistema debe contar con un número telefónico y es dependiente de internet. A pesar de tener algunos límites para el envío de la alarma, estos modos de envío permiten recibir la alarma en cualquier parte del mundo y así reaccionar a esta de manera oportuna. En la Figura 2-7 se observa el diagrama de envío de alarma por medio de alguna de las opciones mencionadas, donde se muestra que se recibe la información de los distintos sensores y luego, en caso de activarse la alarma, se envía por internet o GSM dependiendo el medio de envío, de forma que llegará al computador o teléfono.

Muchos dispositivos de monitoreo tienen la opción de envío de alarma por medio de E-mail o SMS, un ejemplo de envío por E-mail es el dispositivo “SureCross FlexPower” [14], mientras que los dispositivos “Control GSM 5010 5011” realizan envíos de alarma por medio de SMS [10]. Actualmente no hay empresas que apliquen el envío de alarmas por WhatsApp, sin embargo, ya se están realizando proyectos que trabajan con este medio

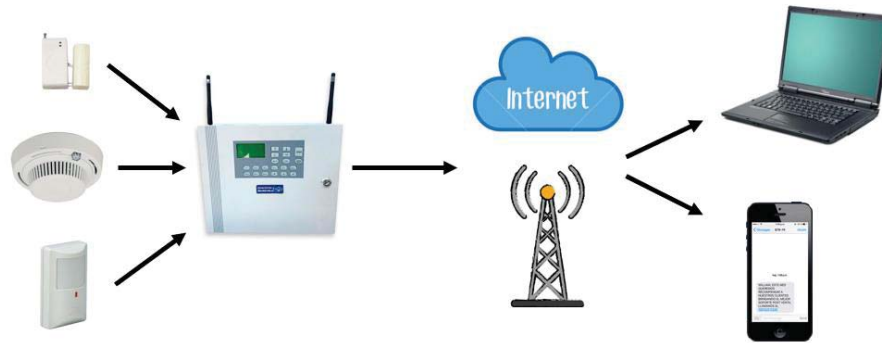


Figura 2-7 Diagrama envío de Alarma vía E-mail, SMS o WhatsApp

2.2.3 Comparación Sistemas de Alarma

En general, los sistemas de monitoreo tienen incorporadas alarmas, donde las más utilizadas son vía E-mail y SMS, ya que permiten obtener las alarmas en cualquier parte que se encuentre el cliente y así poder realizar alguna acción. En la Tabla 2-2 se realiza la comparación de las distintas formas de envío de la alarma.

Tabla 2-2 Tabla comparativa distintos sistemas de alarma

	Alarma por medio de sirena y/o luz	Alarma vía E-mail	Alarma vía SMS	Alarma vía WhatsApp
Alcance	En Lugar	El mundo	País y/o Mundo	Mundo
Dependencia Internet	NO	SI	NO	SI
Requiere SIM con dinero	NO	NO	SI	NO
Costo (alto, medio, bajo)	BAJO	MEDIO	MEDIO	-

La alarma a través de sirena o luz no son muy útiles en invernaderos, ya que implicaría tener una persona constantemente atenta por si se activa la alarma. Si bien la alarma vía WhatsApp no se ha implementado aún en empresas dedicadas a este rubro, es una opción interesante debido a que todas las personas se comunican a través de esta aplicación. Las más utilizadas en el mercado son vía E-mail y SMS debido a su largo alcance y comodidad de recepción de la alarma.

2.3 Empresas de Sistemas de Monitoreo y Alarma en Chile y el Mundo

Hoy en día existen diversas empresas dedicadas a ofrecer servicios de monitoreo y alarma alrededor del mundo, algunas de ellas están insertas en el mercado desde los años 90 o incluso antes, donde la tecnología aún no era muy avanzada. Algunas de estas empresas son pioneras en esta área, comenzando como pequeñas empresas y hoy son reconocidas mundialmente. A nivel nacional, esta área de la tecnología ya está avanzada y hay varias empresas que ofrecen este servicio. A continuación, se mencionarán algunas empresas en Chile y el mundo.

2.3.1 Meldic, Chile

Fundada en 1987 para desarrollar y fabricar instrumentos de medición, donde se incluyen los sistemas de monitoreo, los cuales se han ido potenciando en el tiempo incorporando alarmas mediante SMS. Esta empresa ofrece una amplia gama de instrumentos y equipos de medición e inspección de última generación para las áreas de la industrial, medioambiental, médica y laboratorios. (Figura 2-8)



Figura 2-8 Logo Meldic (fuente: www.meldic.cl)

2.3.2 Unisource, Chile

Se fundó en 1996, liderando en el país los dataloggers en las áreas de transporte y exportación, industria láctea, agroindustria, avícola y acuícola. Son expertos en monitoreo integral con sensores, registradores de datos y telemetría, además, incorporan alarmas vía SMS o E-mail a sus sistemas. (Figura 2-9)



Figura 2-9 Logo Unisource (fuente: www.unisource.cl)

2.3.3 Epssilon, Chile

Fundada en 2011, se especializa en sistemas de monitorización remota, con aplicaciones en distintos ámbitos y sectores industriales, mediante cableado o WSN (redes de sensores inalámbricos), y con opción a incorporación de alarma vía SMS. (Figura 2-10)



Figura 2-10 Epssilon (fuente: www.epssilon.cl)

2.3.4 Banner Engineering, EEUU

Fundada en 1966, la empresa estadounidense comenzó como una pequeña empresa de ingeniería conocida por resolver problemas. Hoy en día es una gran empresa líder mundialmente reconocido en el área de automatización industrial y de procesos. Cuenta con soluciones para agricultura de monitoreo de temperatura y humedad, con envío de mensajes de estado o de alerta por correo electrónico. (Figura 2-11)



Figura 2-11 Logo Banner Engineering (fuente: www.bannerengineering.com)

2.3.5 Etherpower, EEUU

Con más de 10 años de experiencia, provee productos novedosos que permitan de forma remota el acceso, monitoreo, alarmas automáticas y control de ambientes, instalaciones y equipos. Tiene sedes en Estados Unidos, Argentina y México (Figura 2-12)



Figura 2-12 Logo EtherPower (fuente: www.etherpower.net)

2.3.6 Novus, Brasil

Se fundó en 1982 en Brasil, dedicada al desarrollo, fabricación y comercialización de instrumentos electrónicos de medición y control, principalmente de temperatura, con aplicaciones en laboratorios e industrias. Algunos de sus instrumentos cuentan con envío de alarmas vía SMS. (Figura 2-13)



Figura 2-13 Logo Novus (fuente: www.novusautomation.com)

2.3.7 Resumen Empresas

Todas las empresas mencionadas anteriormente se dedican a ofrecer servicios de sistemas de monitoreo y alarma de distintos tipos. En la Tabla 2-3 se puede ver el cuadro de resumen de estas empresas, mencionando el sistema de monitoreo y alarma principal que ofrecen.

Tabla 2-3 Cuadro de resumen empresas

	Año	País	Sistema de monitoreo	Sistema de alarma
Meldic	1987	Chile	Nube	SMS
Unisource	1996	Chile	En Lugar	E-mail y SMS
Epssilon	2011	Chile	IP	SMS
Banner Engineering	1966	EEUU	E-mail	E-mail
Etherpower	2005	EEUU	IP	E-mail y SMS
Novus	1982	Brasil	Nube	SMS

2.4 Proyecto de medición de temperatura en sala de la EIE

Existe un proyecto de un alumno de la Escuela de Ingeniería eléctrica donde el objetivo principal consistía en monitorear la temperatura de la sala de redes de la Escuela de Ingeniería Eléctrica. Para esto se relacionaron los módulos XBee con la placa Raspberry Pi, donde la comunicación se logra utilizando una placa Arduino Uno como puente entre ambos.

El proyecto se compone de dos módulos XBee, conectado cada uno a un sensor de temperatura LM35DZ, que funcionan como Router, uno dentro de la sala de redes y otro fuera. Estos dos transmiten la información de los sensores hacia un módulo XBee funcionando como

coordinador, el cual está conectado al Arduino, este último lee la información recibida y la transforma a un valor en grados Celsius. Luego, esta información transformada se envía por puerto serial hacia la Raspberry Pi para luego poder visualizarla en una página web. El XBee coordinador está en modo API para obtener una interfaz estructurada donde los datos se envían a través de tramas. Los otros dos XBee Router están en modo Transparente y a cada uno se le activa el pin análogo necesario para recibir la información de los sensores de temperatura. En la Figura 2-14 se observa el diagrama de conexiones utilizado para la realización del monitoreo de temperatura.

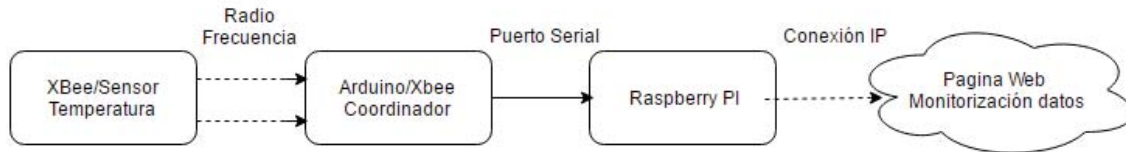


Figura 2-14 Diagrama representativo del proyecto [15]

Una vez que el Arduino recibe la información, debe ser procesada para luego enviarla a la Raspberry Pi. El Arduino lee la información recibida por el XBee, reconoce de qué módulo viene y convierte el dato análogo de los sensores a grados Celsius para finalmente mostrarlo en pantalla. El monitoreo de la temperatura se visualiza a través de una página web, de forma que cualquier dispositivo conectado a la misma red de internet que la Raspberry Pi podrá visualizar los datos. Para hacer la página, se utiliza Tornado-Serial, que es un framework que maneja comunicaciones que son recibidas por el puerto serial de la Raspberry Pi y WebSocket que la aplicación que sube la información que llega por el puerto serial a una dirección web [15]. Finalmente, la página web queda como se observa en la Figura 2-15, la cual muestra dos gráficos, el primero muestra la temperatura dentro de la sala de Redes y el segundo la temperatura fuera de la sala de redes. Además, en la parte superior muestra el último mensaje recibido, es decir, el último valor en grados Celsius que se recibió y desde que módulo viene (A o B).

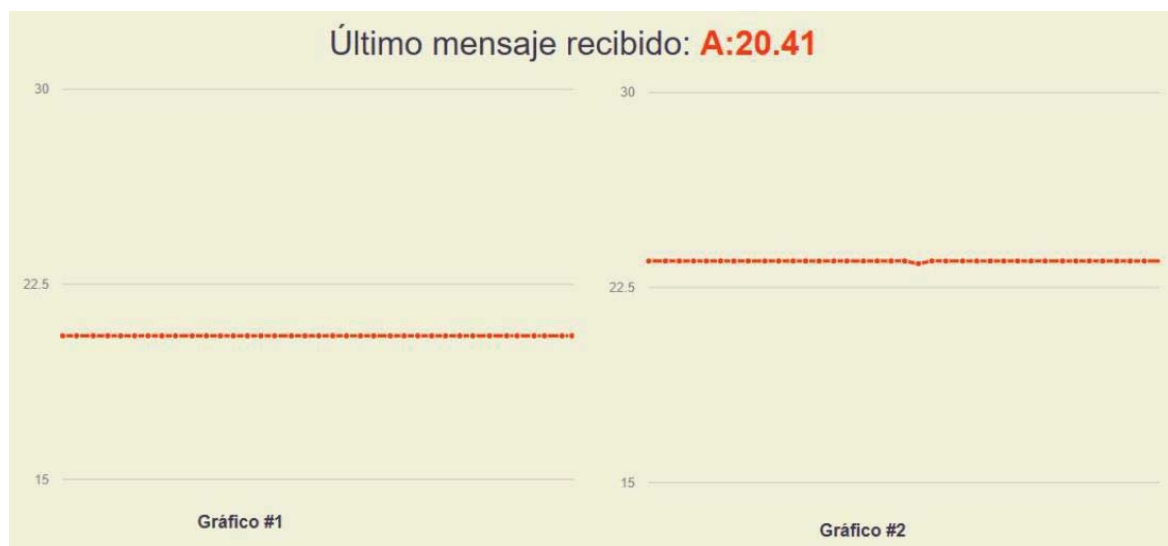


Figura 2-15 Página web obtenida en proyecto de la EIE [15]

3 Marco Teórico

3.1 Descripción de Hardware

3.1.1 XBee

Los módulos XBee son pequeños chips compuestos por un microcontrolador, un emisor y un receptor de radio frecuencia. Estos módulos brindan un medio inalámbrico para la interconexión y comunicación entre dispositivos. Utilizan un protocolo de red llamado IEEE 802.15.4 para crear redes fast point-to-multipoint (punto a multipunto); o para redes peer-to-peer (punto a punto). Fueron diseñados para aplicaciones que requieren de un alto tráfico de datos, baja latencia y una sincronización de comunicación predecible. Son propiedad de Digi basado en el protocolo Zigbee (estándar para el armado de redes inalámbricas a corta distancia y baja velocidad de datos) [16]

Estos módulos son fáciles de usar, ya que no requiere de grandes conexiones para funcionar, solo se necesitan las conexiones: voltaje de alimentación (3.3 V), tierra (GND) y transmisión (TX) y recepción (RX) de datos. Además, son altamente configurables y soportan múltiples protocolos para permitir su uso desde un enlace punto a punto hasta una gran red malla. Así, se puede conectar un microcontrolador y el XBee directamente, obteniendo una comunicación inalámbrica de las aplicaciones generadas por el microcontrolador. En la Figura 3-1 se puede observar la vista superior de un módulo XBee, donde se muestran los pines de alimentación, tierra, transmisión, recepción y entradas/salidas.

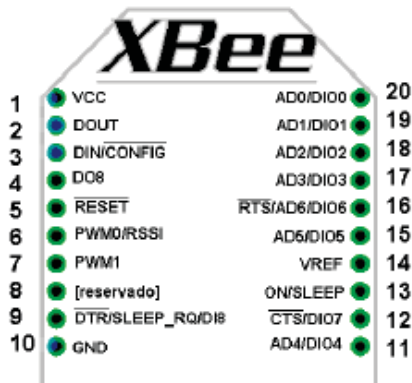


Figura 3-1 Vista superior de diagrama de pines de módulo XBee (fuente: www.mcielectronics.cl)

Los XBee tienen un conjunto de pines de entradas y salidas (Tabla 3-1), donde se pueden conectar sensores o actuadores y configurarlos para un comportamiento específico. Tiene entradas y salidas tanto analógicas como digitales [17], estas últimas sólo permiten valores de 0 o 1 (Bajo o Alto), las cuales se podrían utilizar en aplicaciones ON-OFF, por lo tanto, no podrá leer valores de sensores digitales debido a que entregan una serie de bits.

Tabla 3-1 Entradas y salidas de módulos XBee

Pin	Entradas / salidas (I/O)
6	PWM 0
7	PWM 1
11	I/O digital 4
12	I/O digital 7
15	I/O digital 5
16	I/O digital 6
17	I/O digital 3 / entrada analógica 3
18	I/O digital 2 / entrada analógica 2
19	I/O digital 1 / entrada analógica 1
20	I/O digital 0 / entrada analógica 0

Existen diversos modelos que varían según los requerimientos necesarios, pero todos cumplen la misma función principal. Estos se pueden agrupar en dos tipos, serie 1 (topología estrella) y serie 2 (topología de malla); son muy similares en cuanto a hardware, pero no son compatibles entre sí, la elección entre estas dos depende de las aplicaciones que se le asignarán. Además, de cada modelo de XBee existe la versión regular y la versión PRO; la versión PRO tiene mayor capacidad de alcance y puede doblar la distancia de transmisión y tienen mayor consumo.

Por otra parte, cada módulo permite emplear distintos tipos de antenas (Figura 3-2): Chip antena, es un chip cerámico que permite una radiación unidireccional, irradia o capta desde la parte frontal; Wire antena, es un cable que sobresale del módulo y permite una radiación omnidireccional, es decir, hacia todas las direcciones; conector U. FL, es un conector miniatura para radio frecuencias que alcanza hasta los 6 [GHz]; conector RPSMA, es una variante del conector U. FL más grande y voluminosa [16].

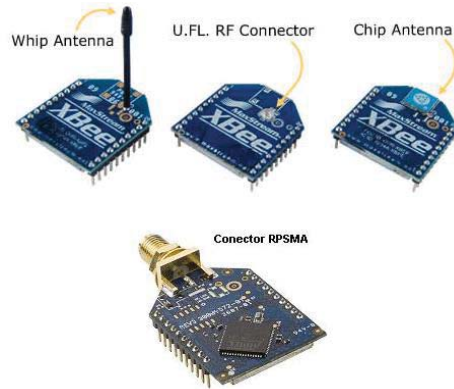


Figura 3-2 Tipos de antenas de módulos XBee (fuente: www.xbee.cl)

Existen varias formas de clasificar los XBee según ZigBee, como lo son los tipos de dispositivos, las topologías de red y los modos de comunicación. En la Tabla 3-2 se pueden observar las distintas clasificaciones con sus respectivas descripciones.

Tabla 3-2 Clasificación de XBee

Clasificación	Descripción	
Tipos de dispositivos	Coordinador	Controla el ruteado y administra la red
	Router	Interconecta diferentes nodos
	Dispositivo Final	Responde a peticiones de otros dispositivos
Topologías de red	Punto a punto	Un dispositivo es el coordinador y el otro puede ser un router o dispositivo final
	Estrella	Todos los dispositivos de la red se pueden comunicar con el coordinador
	Malla	Cualquier dispositivo se puede conectar con otro y el coordinador
	Árbol	Coordinador es la red inicial, router forma las ramas y dispositivo final las hojas del árbol
Modos de comunicación	Transparente	Información pasa exactamente como la recibe por puerto serial
	API	Transmite los datos de forma estructurada

El modo API (Application Programming Interface) (Figura 3-3) es más complejo que el modo transparente, pero ofrece mayores posibilidades en el control de la red. Este modo tiene varias ventajas en comparación al modo transparente [18], tales como:

- Comunicar con uno o varios destinatarios
- Identificar de qué módulo provienen los datos
- Recibir el estado de la transmisión de paquetes
- Hacer gestión y diagnóstico de la red

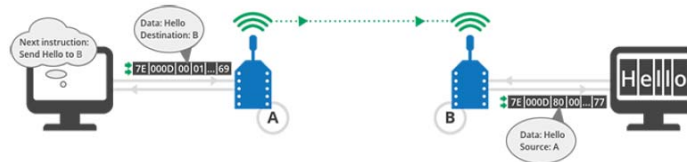


Figura 3-3 Envío de datos en Modo API (fuente: www.digi.com)

En modo API los datos están estructurados en tramas. Se envían y reciben los datos a través de la interfaz serie y contienen el mensaje inalámbrico propio y, además, información adicional como fuente/destino, calidad de la señal, entre otros [18]. En la Figura 3-4 se observa cómo están estructuradas las tramas.

Start delimiter	Length		Frame data								Checksum
	2	3	4	5	6	7	8	9	...	n	
0xE7	MSB	LSB	API-specific structure								Single byte

Figura 3-4 Tramas API (fuente: www.digi.com)

- **Start delimiter:** Cada trama API comienza con un byte de inicio para indicar que se está al comienzo de una trama de datos, su valor siempre es 0xE7
- **Length:** Los dos bytes siguientes indican el tamaño en bytes de la trama de datos, de esta forma se conoce la longitud que se debe leer para obtener la información
- **Frame Data:** Es específica de cada mensaje recibido o enviado, ya que es la información propia que se ha recibido o enviado
- **Checksum:** Es el último byte de la trama, sirve para comprobar la integridad de la información recibida o enviada. Se calcula sumando todos los bytes de la trama, excepto los delimitadores y la longitud, y quedándose con los 8 bits menos significativos del resultado total.

Con esta estructura se logra determinar quién envía y quién recibe la información, además comprueba si esta llega de manera correcta. Es por esto que, para proyectos de mayor complejidad, donde se necesita saber cómo está llegando la información para luego poder procesarla, se utiliza el modo API.

Estos dispositivos pueden operar de forma independiente pero solo pueden realizar el envío de datos, no pueden gestionar los datos enviados o recibidos, sin embargo, pueden comunicarse con otros dispositivos por medio de la interfaz serie. Transmiten al aire los datos que llegan del puerto serie y transmiten al puerto serie cualquier dato que llega por el aire. Por lo tanto, para controlar y gestionar los datos que se envían, se utilizan los microcontroladores. De esta forma, se tienen dos tipos de comunicación en los módulos XBee (Figura 3-5): comunicación inalámbrica, que es la comunicación entre módulos pertenecientes a la misma red, y comunicación serie, que es la comunicación entre un módulo y un microcontrolador o PC [19].



Figura 3-5 Tipos de comunicación de módulos XBee (fuente: www.digi.com)

Para que dos o más módulos interactúen entre sí, es necesario que estén en la misma frecuencia (Canal) y en la misma red (Identificador de red de área personal) [19]. En la Figura 3-6 se observa la separación de red por ID y canal.

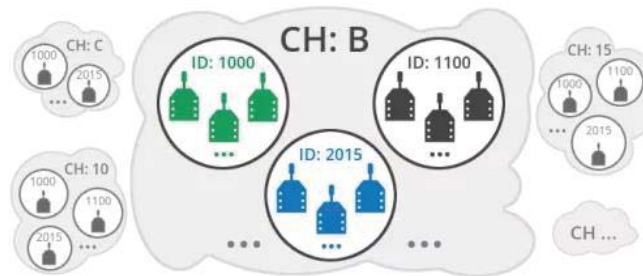


Figura 3-6 (fuente: www.digi.com)

Por otra parte, cada módulo XBee tiene una dirección MAC única de 64 bits, la cual se compone de los parámetros Serial Number High (SH) y Serial Number Low (SL). SH generalmente es igual para todos (0013A200) e identifica a los módulos de Digi.

Para la realización de este proyecto se utiliza el XBee ZigBee Mesh Kit (Figura 3-7) [18], el cual incluye:

- 3 placas de desarrollo XBee Grove
- 3 módulos XBee ZigBee con antena integrada. Con rango urbano de 60 m y línea vista de 1200 m.
- 3 cables micro USB



Figura 3-7 Xbee ZigBee Mesh Kit (fuente: www.digi.com)

3.1.2 Raspberry Pi

Raspberry Pi es un micro ordenador de placa reducida o de placa única y de bajo costo, desarrollado en Reino Unido por la Fundación Raspberry Pi, con el objetivo de obtener un dispositivo de bajo costo que permitiera acercar a todos los públicos un ordenador [20], lo cual permitiría estimular la enseñanza de ciencias de computación en las escuelas. A pesar de su reducido tamaño, este dispositivo puede cumplir las mismas funciones que un PC como, por ejemplo, procesamiento de texto y datos, reproducción de videos y conexión de internet vía Ethernet o wifi.

Este micro ordenador utiliza una tarjeta micro SD para el almacenamiento permanente, el cual funciona como disco duro. En la tarjeta micro SD, se instala el sistema operativo a utilizar, generalmente Linux, en este sistema operativo hay que programar para habilitar los buses y GPIO. Además, se debe alimentar con +5 [V] a 2[A] a través del conector hembra micro USB.

Existen varios modelos de esta placa, los cuales poseen características diferentes, pero su función principal será la misma. En este caso se utilizó la Raspberry Pi 2 modelo B, la cual se muestra en la Figura 3-8.



Figura 3-8 Raspberry Pi 2 Model B (fuente: www.raspberrypi.org)

La placa anterior tiene las siguientes características principales [21]:

- Procesador Broadcom BCM2836 de 900 MHz ARM Cortex-A7 de cuatro núcleos con GPU VideoCore IV de doble núcleo.
- GPU proporciona una tecnología Open GL ES 2.0 hardware acelerado OpenVG y admite imágenes de alta resolución 1080p30 H.264.
- GPU tiene capacidad de 1 Gpixel/s, 1,5 Gtexel/s o 24 GFLOPs con filtrado e infraestructura DMA.
- SDRAM LPDDR2 de 1 GB.
- Dimensiones: 86x56x20 mm.

Una característica muy importante en la Raspberry pi es la fila de pines GPIO (General Purpose Input Output). Esta permite la interacción con el mundo exterior y se encuentra en forma de 40 pines en la Raspberry pi 2 modelo B [22], que es la que se utilizará para este proyecto. Cualquiera de los pines GPIO se pueden utilizar como un pin de entrada o de salida y sus utilidades son múltiples. Un detalle importante de estos pines es que son del tipo ‘unbuffered’, es decir, no tienen buffers de protección, por lo tanto, hay que tener en cuenta las magnitudes que se conectarán a los pines para no dañar la Raspberry Pi. Otro detalle a considerar es que son puertos completamente digitales, por lo tanto, solo detectará niveles altos y bajos y no se podrán leer señales análogas a no ser que se utilice un convertor análogo-digital.

No todos los pines del puerto GPIO son destinados a entradas y salidas. Están los pines de alimentación, los cuales sirven para alimentar diferentes circuitos o componentes, estos pines son de 3.3 V, 5V y tierra. También hay otros pines especiales, los cuales están destinados a una interfaz UART, conexiones Tx y Rx para comunicaciones serie, buses de comunicación, entre otros. Y están los pines normales, que son para entradas y salidas y se configuran a través de software. En la Figura 3-9 se puede observar el esquema de conexión de los pines GPIO, donde en amarillo se muestran los pines destinados a entradas y salidas [22].

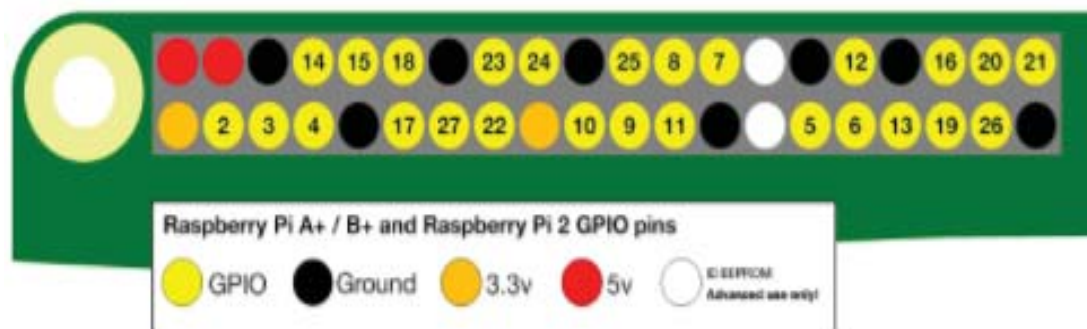


Figura 3-9 Esquema GPIO Raspberry Pi 2 modelo B (fuente: www.raspberrypi.org)

3.1.3 Arduino

Arduino es un microcontrolador de código abierto basada en hardware y software flexibles y fáciles de usar. Puede recibir entradas desde una variedad de sensores y como salida puede controlar luces, motores y muchos otros artefactos. Se programa utilizando Arduino Programming Language y Arduino Development Environment [23].

Esta plataforma tiene diversos beneficios en comparación a otros microcontroladores, algunos beneficios son: Multiplataforma, se ejecuta en sistemas operativos Windows, Macintosh OSX y GNU/Linux, a diferencia de la mayoría de los microcontroladores que están limitados a Windows. Código abierto, el código se puede modificar completamente, puede ser expandido mediante librerías C++ AVR C. Hardware extensible, los diseñadores de circuitos pueden hacer su propia versión de Arduino, extendiéndolo y mejorándolo.

Existe una gran variedad de Arduino, cada uno con distintas características y componentes dependiendo de lo que se requiera, pero el objetivo principal siempre es el mismo. En el proyecto se utilizará el Arduino Uno R3, el cual se puede ver en la Figura 3-10.



Figura 3-10 Arduino Uno R3 (fuente: www.arduino.cc)

Arduino Uno R3 tiene las siguientes características [24]:

- Microcontrolador ATmega328.
- Voltaje de entrada 7-12 V.
- 14 pines digitales de I/O (6 salidas PWM).
- 6 entradas análogas.
- 32k de memoria Flash.
- Reloj de 16 MHz de velocidad.

3.1.4 Sensores

Los sensores son dispositivos eléctricos y/o mecánicos que interactúan entre la tecnología y el entorno. Son diseñados para recibir información desde el exterior, donde obtiene valores de magnitudes físicas, para luego transformarla en una magnitud que se pueda cuantificar y manipular [25]. El proceso comienza captando la magnitud física y luego muestra en la salida una señal eléctrica dependiente del valor de la variable física, luego, la señal eléctrica es modificada por un sistema de acondicionamiento de señal y se obtiene una salida en voltaje, finalmente, el sensor transforma y/o amplifica la tensión de salida y pasa a un conversor A/D conectado a un PC, el conversor transforma la señal de tensión continua en una señal discreta. De esta forma se puede visualizar y manipular la información obtenida de los sensores.

Existe una gran cantidad de sensores, algunos ejemplos son sensor de corriente, de humedad, de presión, de proximidad, de temperatura, de velocidad y muchos otros. Sin embargo, en este proyecto hay cuatro sensores de gran importancia: sensor de temperatura, de humedad, de dióxido de carbono y de luminosidad.

3.1.5 DHT 22

El sensor DHT22 (Figura 3-11) es un sensor digital de temperatura y humedad, el cual utiliza un sensor capacitivo de humedad y un termistor para la temperatura. Este sensor muestra los datos a través de una señal digital en el pin de datos, el cual es de 40 bits, los 16 primeros bits son de humedad relativa, los siguientes 16 bits son de temperatura y los últimos 8 bits son para corroborar que se envió el dato. Este posee las siguientes características [26]:

- Alimentación: $3.3 \text{ Vdc} \leq V_{cc} \leq 6 \text{ Vdc}$.
- Rango de temperatura: $-40 - 80^\circ\text{C}$.
- Precisión de temperatura: $< \pm 0.5^\circ\text{C}$.
- Resolución de temperatura: 0.1°C .
- Rango de humedad: $0 - 100\% \text{ RH}$.
- Precisión de humedad: $2\% \text{ RH}$.
- Resolución de humedad: $0.1\% \text{ RH}$.
- Tiempo de censado: 2 segundos.

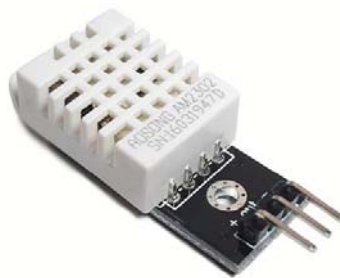


Figura 3-11 Sensor DHT22 (fuente: www.ebay.com)

3.1.6 BH-1750

El sensor BH-1750 (Figura 3-12) es un sensor digital que mide el nivel de luz del ambiente, tiene incorporado un convertidos A2D de 16 bits para emitir directamente una señal digital por medio del bus I2C, y está diseñado para tener una respuesta espectral similar a la del ojo humano, por lo que proporciona la medición directamente en Lux. Este sensor posee un amplio rango de medición ajustable, por lo que es capaz de medir en casi cualquier situación de iluminación. Este sensor posee las siguientes características [27]:

- Voltaje de operación: 2.4 - 3.6Vdc.
- Rango de medición: 1 - 65.535lux.
- Rechazo de ruido a 50/60 Hz.
- Consumo promedio: 140 - 199 μ A.
- Temperatura de operación: -40 - 85°C.



Figura 3-12 Sensor BH-1750 (fuente: www.ebay.com)

3.1.7 MG-811

El sensor MG-811 (Figura 3-13) posee una capa electrolítica cubierta por una malla de acero, el cual permite detectar el dióxido de carbono del ambiente. Este es un sensor análogo que generalmente tiene incorporado un módulo que tiene un circuito que permite amplificar la señal o ajustar la sensibilidad, esto permite que el sensor se adapte a los distintos ambientes que se pueden tener. Se obtienen mediciones entre 30 - 50mV, lo que es equivalente a 350 - 10.000 partes por millón de CO₂ en el aire. Este sensor tiene as siguientes características [28]:

- Voltaje operativo: < 5.5V.
- Corriente de operación: 200mA.
- Temperatura de operación: -20 - 50°C.
- Humedad de operación: < 65%.
- Rango de medición: 350 - 10.000ppm.
- Tiempo de respuesta: < 5 segundos.



Figura 3-13 Sensor MG-811 (fuente: www.ebay.com)

3.2 Descripción de Software

3.2.1 X-CTU

X-CTU es un software multiplataforma ofrecido por Digi Internacional el cual permite interactuar con los módulos XBee de manera fácil, visual y rápida. Este se puede descargar gratuitamente desde la página de Digi. Este software incluye herramientas que permiten configurar y probar los módulos de manera muy simple. Este software permite configurar, inicializar, actualizar firmware y testear los módulos XBee, comunicándose por puerto serie a los módulos [19].

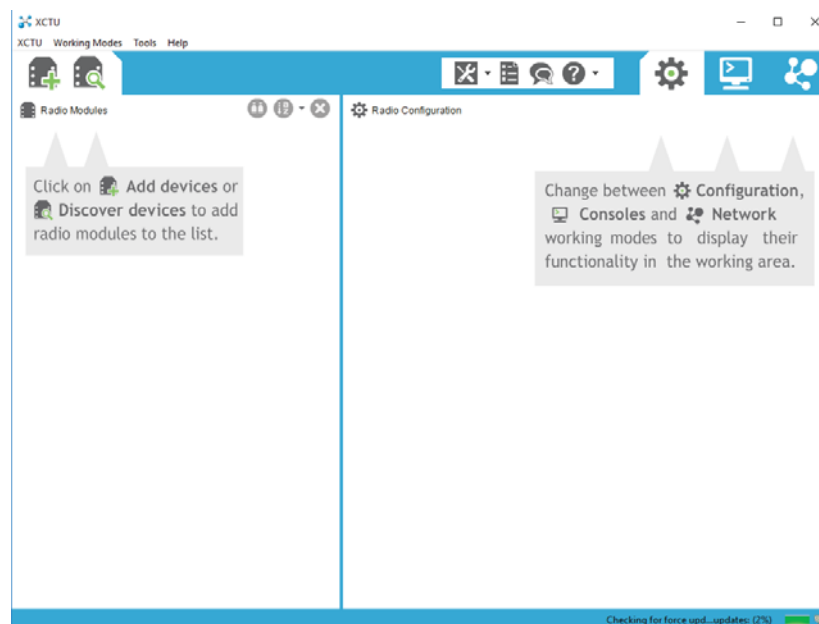


Figura 3-14 X-CTU

En la Figura 3-14 se muestra la inicialización del software, aquí se pueden buscar y seleccionar los módulos para luego poder configurarlos. Una vez configurados, se puede probar si hay comunicación entre los módulos a través de la consola, donde se permite el envío de mensajes o datos entre ellos.

3.2.2 Arduino IDE

Arduino IDE (Integrated Development Environment) (Figura 3-15) es el software donde se desarrolla el código para cualquier dispositivo Arduino, está escrito en Java y está basado en Processing y otro software de código abierto [29]. Tiene una interfaz sencilla y tiene la capacidad de expansión de librerías para prácticamente cualquier dispositivo electrónico.

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Además, Arduino incorpora herramientas para cargar el programa ya compilado en la memoria flash del hardware.

En esta plataforma se desarrolla toda la programación necesaria para lograr la relación que se requiere entre el Arduino y los componentes electrónicos, ya sean sensores, actuadores e incluso los propios módulos XBee.

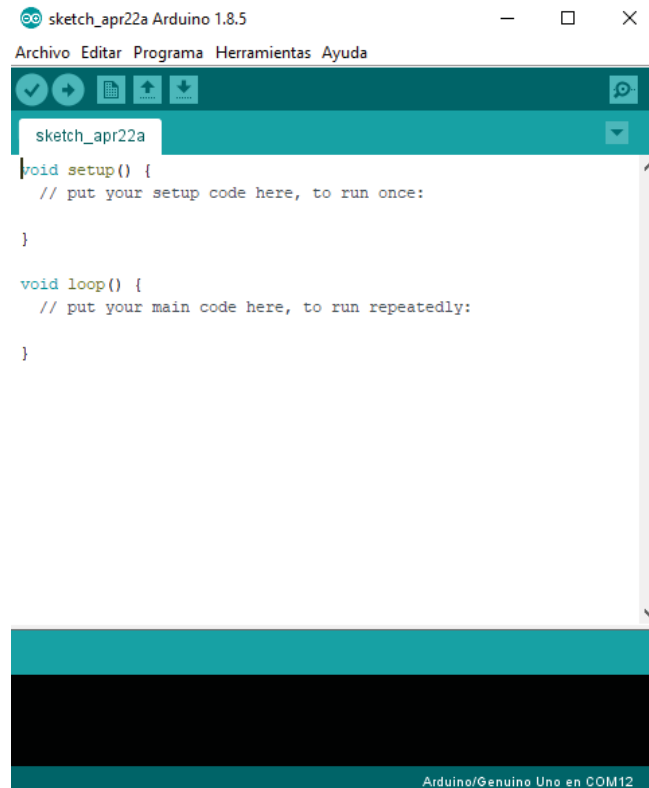


Figura 3-15 Arduino IDE

3.2.3 Python

Python (Figura 3-16) es un lenguaje de programación de código abierto creado a principios de la década de los noventa por el europeo Guido Van Rossum con el fin de obtener un lenguaje multiparadigma, es decir, que soporte programación orientada a objetos, programación estructurada y programación funcional, y que sea sencillo, ágil y fácil de aprender. Además, puede ser utilizado en diversas plataformas y sistemas operativos, como Windows, Mac OS o Linux. Este es un lenguaje de scripting, lo que significa que no se compila debido a que utiliza un intérprete y muchas veces permite ahorrar tiempo en la creación de programas [30].

Gracias a que Python es un lenguaje multiparadigma, se puede utilizar para el desarrollo de diversas aplicaciones, como software para aplicaciones científicas, comunicaciones de red, aplicaciones de escritorio con interfaz gráfica (GUI), aplicaciones web, creación de juegos, entre otros.



Figura 3-16 Logo Python (fuente: www.python.org)

3.2.4 MySQL

Es un sistema de gestión de base de datos relacional de código abierto basado en el lenguaje de consulta estructurado (SQL), originalmente creado por la empresa MySQL AB y actualmente pertenece a Oracle Corporation. Este gestor es multihilo y multiusuario, lo cual le permite ser utilizado y realizar consultas por varias personas al mismo tiempo [31]. Es de gran utilidad para aplicaciones web donde se deben mantener los datos organizados. Se permite acceder a ella desde varios lenguajes de programación como C, C++, JAVA, PHP, Python, etc.

MySQL (Figura 3-17) se utiliza en base de datos que requieren centralizar información, con datos que se pueden estructurar en tablas con campos definidos, índices y llaves que relacionan las tablas entre sí. Este trabaja con un sistema centralizado de gestión de datos, por lo que permite realizar cambios en un solo archivo y que se ejecute en toda la estructura de datos que se comparte en la red, de esta forma se evita tener que modificar todo el código web.



Figura 3-17 Logo MySQL (fuente: www.mysql.com)

3.2.5 phpMyAdmin

phpMyAdmin (Figura 3-18) es un software de código abierto creado en 1998 para manejar la administración y gestión de base de datos tipo MySQL a través de una interfaz gráfica de usuario, escrito en PHP y ejecutado por internet [32]. En este se puede crear, modificar o eliminar bases de datos, así como las tablas, campos e índices de las mismas, además, se puede importar o exportar una base de datos.

Esta herramienta permite la gestión de MySQL de manera visual e intuitiva, de manera que cualquier persona pueda gestionar una Base de datos sin la necesidad de grandes conocimientos en el tema. Al ser ejecutado vía web, se puede acceder a este desde cualquier dispositivo con conexión al servidor, por lo que se hace aún más fácil para el cliente la manipulación de esta herramienta.



Figura 3-18 Logo phpMyAdmin (fuente: www.phpmyadmin.net)

3.2.6 PHP

PHP (Hypertext PreProcessor) (Figura 3-19) es un lenguaje de programación de código abierto creado en 1995, este lenguaje se especializa en el desarrollo web debido a que se puede incrustar en un archivo HTML, es decir, se puede combinar código PHP con código HTML [33]. Este lenguaje se ejecuta en el servidor, genera HTML y lo envía al cliente, está compuesto de un intérprete encargado de analizar el script que se ha solicitado. Es un lenguaje de gran simplicidad, pero ofrece muchas características avanzadas.

PHP puede ser ejecutado en la mayoría de los sistemas operativos como Linux, Mac OS o Windows, y puede interactuar con los servidores web más populares como Apache. Este lenguaje posee grandes similitudes con los lenguajes comunes de programación estructurada como C y Perl, por lo que se podrán crear aplicaciones más complejas sin la necesidad de aprender mucho más.



Figura 3-19 Logo PHP (fuente: www.php.net)

3.2.7 Apache

Apache (Figura 3-20) es un servidor web de código abierto basado en HTML, este es multiplataforma y soporta diversos sistemas operativos como Windows o Linux. Un servidor web se encarga de procesar una aplicación del lado del servidor, realizando conexiones con el cliente y generando o cediendo una respuesta [34]. Es extensible a través de un sistema de módulos cargables dinámicamente y puede procesar una gran cantidad de idiomas. Este servidor es muy utilizado debido a su flexibilidad, potencia y soporte generalizado.

Este servidor web fue desarrollado en 1995, siendo uno de los más antiguos y más utilizado en el mundo, sin embargo, los últimos años ha descendido la cantidad de usuarios. Suele usarse en conjunto con MySQL y los lenguajes de programación PHP, Perl, Python o Ruby, generando la popular plataforma de aplicaciones LAMP.

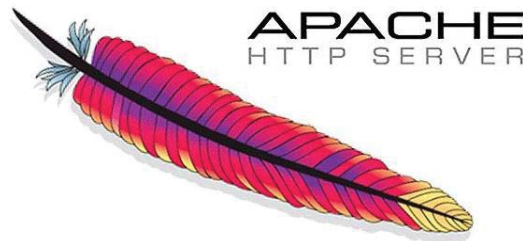


Figura 3-20 Logo Apache (fuente: www.apache.org)

4 Implementación

4.1 Invernadero

Es de gran importancia conocer el medio donde se instalará el proyecto, el invernadero. El invernadero es una estructura cerrada que se utiliza en agricultura para la producción y/o protección del cultivo. Al ser un lugar cerrado, tiene la capacidad de generar condiciones ideales de distintas variables, de forma de producir en temporadas donde no se produce o en sectores donde las condiciones climáticas son muy adversas, debido a que permite el control de las variables. Las variables de mayor importancia en el cultivo son la temperatura, humedad, luminosidad y dióxido de carbono. Al controlar en conjunto estas variables, se pueden obtener condiciones climáticas ideales para la producción, favoreciendo significativamente el desarrollo de las plantas.

Los invernaderos pueden ser de variados tamaños y materiales, por lo que las condiciones serán distintas dependiendo de esto. La estructura puede ser metálica o de madera y la cubierta de plástico o vidrio, lo cual influye en el ambiente que se obtendrá al interior. Además, existen diversos tamaños de invernaderos, donde el tamaño estándar es de 7.5x30m, sin embargo, lo más importante en el tamaño es la altura, debido a que mientras mayor altura se tiene más homogéneo es el ambiente. Al momento de la instalación del proyecto se debe tener en cuenta estos factores y evaluar cuáles serán las condiciones de un invernadero específico para así obtener valores lo más óptimos posibles. En general, los invernaderos tienen climas con alta temperatura y humedad, por lo que se genera mucha transpiración al interior. Se debe evaluar bien la ubicación de los sensores al interior del invernadero, de forma que se obtengan los valores lo más certero posible y además tener en cuenta que habrá transpiración del material generando goteras.

4.2 Previo a la realización del proyecto

Antes de comenzar la realización del proyecto, se configuraron los distintos dispositivos. Los dispositivos más importantes a configurar son los módulos XBee, ya que tienen distintos modos de configuración y si no quedan bien configurados no se logrará la comunicación inalámbrica. Además, se hacen las configuraciones necesarias para lograr la conexión remota del Raspberry Pi al computador para poder configurarlo de forma más eficiente, sin necesidad de tener teclado y mouse adicionales, para esto se deben realizar algunas configuraciones tanto en el computador que se utilizará como en la Raspberry Pi.

4.2.1 Configuración XBee

Para que los módulos XBee puedan comunicarse entre sí hay que configurarlos, además tienen distintas formas de configuración dependiendo de la comunicación que se requiera. En este proyecto, se tendrán dos módulos “Dispositivo Final” y un módulo “Coordinador”. De esta forma, los “Dispositivo Final” serán los que envíen la información de los sensores hasta el “Coordinador”, el cual debe organizar toda la información para luego llevarla al servidor web y realizar el monitoreo. Es por esto que el coordinador necesariamente debe estar en modo API, de esta forma sabrá desde qué “Dispositivo Final” viene la información y podrá separar cada dato de cada sensor. Los módulos “Dispositivo Final” pueden enviar la información en modo Transparente, ya que solo envía la información, no la procesa.

La configuración de los módulos XBee se hace a través del software X-CTU. Aquí existen tres posibles configuraciones. AP=0, donde se deshabilita la trama API, es decir, se trabaja en modo transparente. AP=1, el módulo trabaja en modo API. AP=2, se trabaja en modo API, pero con carácter de escape [35]. Este último hace que la trama sea más grande, pero evita que se confunda el byte de inicio de la trama con los datos enviados, es por esto que se utilizará este último modo en el módulo “Coordinador”.

Para comenzar con la configuración, se ingresa a X-CTU en el computador y se presiona en “Add devices”, aquí se mostrarán todos los módulos conectados al computador, en este caso, dos. Seleccionar ambos módulos y luego se lee la configuración que tiene cada módulo. Se dejarán configurados de fábrica, para esto se presiona en “Default” y luego presionar “Write” para guardar la configuración. Ahora se pueden configurar ambos dispositivos, uno será configurado como “Coordinador” y el otro como “Dispositivo final”. Los parámetros importantes para cualquier comunicación entre módulos son el ID Network (red de comunicación), NI (identificador de nodo) y AP (AP=0,1,2). Ambos módulos deben tener la misma ID, no importa el número, sino que todos los módulos que se quieren comunicar tengan la misma ID. El NI es para identificar cuál será el “Coordinador” y cuál el “Dispositivo final”. Para el “Coordinador” se dejará AP=2 y para el “Dispositivo final” será AP=0. A continuación se pueden ver los parámetros modificados en ambos dispositivos. Se observa que en el “Coordinador” se modifica el parámetro “Coordinador Enable (CE)”, el cual indica que este módulo trabajará como coordinador, se deja como habilitado. En el “Dispositivo Final” se modifica el parámetro “Join Verification (JV)”, el cual permite que, si en la red hay un coordinador, la información se envíe automáticamente a este [35]. Una vez que se realizaron todas las modificaciones se vuelve a presionar “Write” para guardar esta configuración. En la Figura 4-1 se observa la configuración realizada de los módulos XBee en X-CTU.

i	ID PAN ID	3332
i	SC Scan Channels	1FFE Bitfield
i	SD Scan Duration	4 exponent
i	ZS ZigBee Stack Profile	0
i	NJ Node Join Time	FF x 1 sec
i	NW Network Watchdog Timeout	0 x 1 minute
i	JV Channel Verification	Enabled [1]
i	JN Join Notification	Disabled [0]
i	OP Operating PAN ID	0
i	OI Operating 16-bit PAN ID	FFFF
i	CH Operating Channel	0
i	NC Number of Remaining Children	14
i	CE Coordinator Enable	Enabled [1]
i	BD Baud Rate	9600 [3]
i	NB Parity	No Parity [0]
i	SB Stop Bits	One stop bit [0]
i	RO Packetization Timeout	3 x character times
i	D7 DIO7 Configuration	CTS flow control [1]
i	D6 DIO6 Configuration	Disable [0]
i	AP API Enable	API enabled with escaping [2]
i	AO API Output Mode	Native [0]

Figura 4-1 Configuración módulo XBee

Una vez que se configuraron los módulos, se debe comprobar la comunicación entre estos. Para esto se debe ir a “Consoles”. Aquí se verán algunas diferencias debido a que un módulo está en modo transparente y el otro en API, por lo que el modo transparente recibe la información de la misma forma en que se envía, mientras que el modo API lo hace en tramas. En la Figura 4-2 se puede observar que la prueba de conexión entre ambos módulos fue exitosa, se observa que el envío del mensaje desde el Router1 llegó en forma de trama al Coordinador y el mensaje del Coordinador hacia el Router1 llega de forma normal. Estas diferencias se deben a que se configuró el Router1 en modo transparente y el Coordinador en modo API.

The screenshot displays two windows: 'Console log' and 'Frames log'. The 'Console log' window shows the following text:

```
Saludos desde
CoordinadorSaludo
s desde Router1
```

The 'Frames log' window shows a table of frames:

ID	Time	Le...	Frame
0	19:08:32....	39	Transmit Request
1	19:08:32....	7	Transmit Status
2	19:08:35....	39	Explicit RX Indicator

The 'Frame details' window on the right shows the following information:

- Receive options: 02
- RF data: ASCII HEX, Saludos desde Router1
- Checksum: 50
- Copy packet information button

Figura 4-2 Conexión Router-Coordinador

4.2.2 SSH y VNC

Acceder a la Raspberry Pi por medio de SSH (Secure SHell) o VNC (Virtual Network Computing) no es necesario para el proyecto, sin embargo, facilita el proceso debido a que se pueden realizar todos los procedimientos desde el mismo computador. Estos son servicios de acceso y control remoto que permitirán trabajar desde el mismo computador y no estar físicamente en la Raspberry Pi.

SSH es un protocolo que permite crear conexiones seguras entre dos computadores, en este caso, entre Raspberry Pi y computador. Así, se puede acceder de forma remota a la Raspberry Pi desde el computador. Este protocolo cuenta con una gran seguridad, todos los datos enviados y recibidos durante la conexión se transfieren por medio de encriptación [36].

Para la utilización de este protocolo en Windows se debe descargar un cliente SSH, donde el más utilizado es PuTTY. Antes de configurar el cliente se debe habilitar SSH en la Raspberry Pi. Luego se accede a PuTTY en el computador y se debe escribir la dirección IP de la Raspberry Pi. Si no hay problemas para acceder, se podrá visualizar el terminal de la Raspberry Pi. A continuación, en la Figura 4-3, se observa cómo se configura PuTTY y, en la Figura 4-4, el terminal en el computador.

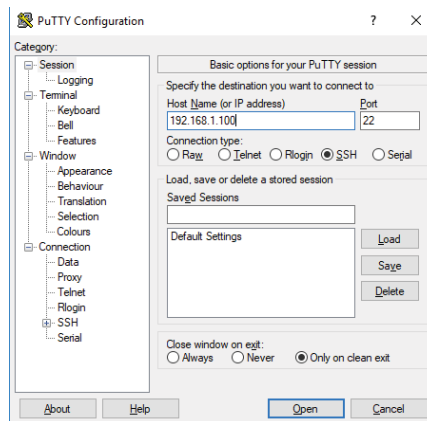


Figura 4-3 SSH con PuTTY

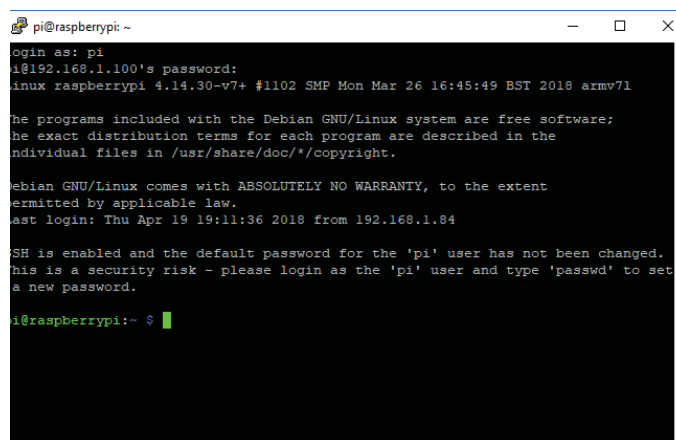


Figura 4-4 Terminal Raspberry Pi en el computador

VNC es un servicio que crea servidores gráficos sobre pantallas virtuales, permite conexiones remotas desde otros computadores de la red al servidor. Es independiente de la plataforma, un cliente VNC puede conectarse a un servidor VNC.

Para la utilización de VNC, se debe instalar el servidor VNC en la Raspberry Pi y el cliente VNC en el computador. Se instaló en el computador “VNC Viewer”. Una vez instalado, se abre VNC Viewer y luego pedirá la dirección IP de la Raspberry Pi, una vez conectado, solicitará el nombre y contraseña que tiene la Raspberry Pi, que por defecto el nombre es pi y la contraseña raspberry. Una vez que se accede, se podrá visualizar la pantalla de la Raspberry Pi en el computador [37], tal como se muestra en la Figura 4-5.

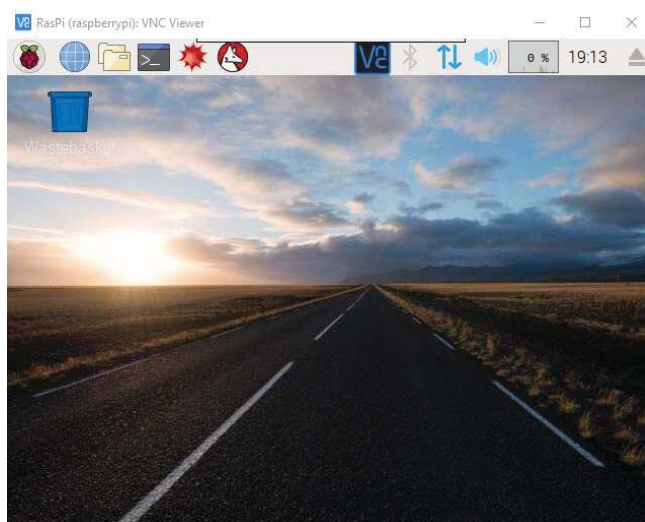


Figura 4-5 VNC

Cabe destacar que la dirección IP de la Raspberry Pi debe quedar fija, ya que para acceder de forma remota pedirá la dirección IP, y si esta es dinámica se tendrá que cambiar cada vez que se quiera acceder. Para este proyecto se prefirió utilizar VNC, ya que es una interfaz gráfica y se puede manejar de mejor forma.

4.3 Diseño Final

Para comenzar la implementación del proyecto, es necesario tener claro el diseño completo, considerando cada nodo y los dispositivos que tiene cada uno. Se debe determinar la cantidad necesaria de nodos “Dispositivo Final” al interior de un invernadero para tener mediciones representativas del lugar, y así tener información lo más certera posible para tomar decisiones.

4.3.1 Dispositivo Final

Como se ha mencionado a lo largo del desarrollo del proyecto, el nodo “Dispositivo Final” es el que se relaciona con el entorno, es decir, está directamente en el invernadero. Este nodo se compone de sensores, Arduino y XBee, donde los sensores obtienen la información del entorno, Arduino se encarga de procesar la información y XBee de enviar la información.

En la Figura 4-6 se puede ver cómo es la conexión de los componentes utilizados en este nodo, donde los sensores y el XBee se conectan al Arduino y todos estos deben ser conectados a una fuente de alimentación que permita el funcionamiento correcto del nodo.

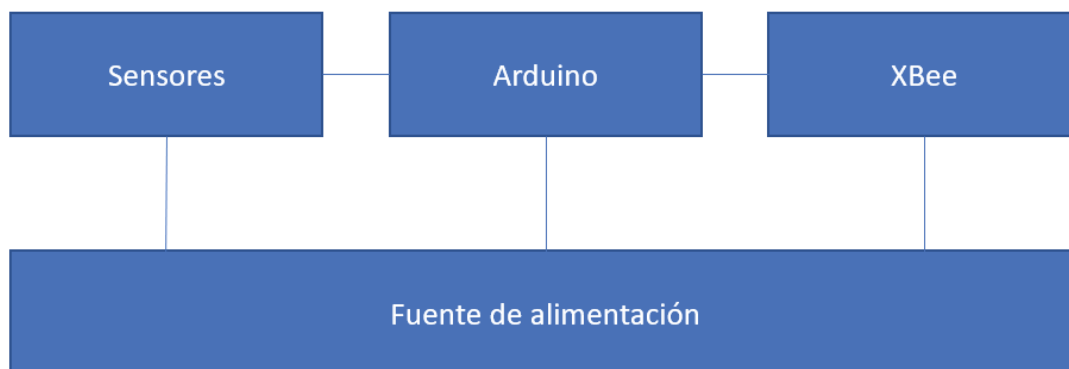


Figura 4-6 Conexión de componentes Nodo “Dispositivo Final”

Se evaluó la posibilidad de quitar el Arduino de este nodo, sin embargo, esto no fue posible debido a que el XBee no es capaz de procesar ni enviar los datos de un sensor digital. Los módulos XBee sólo permiten el envío de información de forma inalámbrica, no puede procesar la información que envía o recibe, para esto es necesario el uso de un microcontrolador. Por otro lado, se ha comentado que los módulos XBee poseen entradas y salidas tanto análogas como digitales, sin embargo, las entradas y salidas digitales solo permiten un 0 o 1 (Bajo o Alto), las cuales se podrían utilizar en aplicaciones ON-OFF, es por esto que el módulo no es capaz de leer la entrada de los sensores digitales, donde el sensor DHT22 entrega un valor de 40 bits y el sensor BH-1750 un valor de 16 bits. Además, el sensor BH-1750 funciona con un bus de comunicación I2C y los módulos XBee no son capaces de transmitir los datos I2C. Por las razones comentadas es que se llegó a la conclusión que el nodo debía tener incorporado un microcontrolador, en este caso, Arduino.

Un elemento de vital importancia en este nodo es la fuente de alimentación debido a que es un nodo inalámbrico y se debe elegir una opción que permita la mayor duración posible sin tener que reemplazarla. Se debe considerar el almacenamiento de la energía y el suministro de esta de la forma necesaria para los dispositivos, además, se debe considerar cómo se recargará la batería o si será reemplazada al fin de su vida útil.

La fuente de alimentación más utilizada es la batería, las cuales existen recargables (baterías secundarias) o no recargables (baterías primarias) dependiendo si tiene algún mecanismo externo de recarga [38]. Las primarias son principalmente de bajas potencias y capacidad, por lo tanto, se descartarán, ya que se requiere la mayor capacidad posible. En cambio, las secundarias tienen un amplio margen de potencia y capacidad, desde unos cientos de mAh hasta capacidades de varias decenas de Ah, por lo que se podría encontrar una batería con harta capacidad para tener mayor duración.

Por otro lado, es importante determinar el consumo aproximado del nodo y así saber cuánto duraría la batería. En la Tabla 4-1 se muestra el consumo de cada dispositivo, considerando un consumo total de los sensores de 50 mA, y el tiempo que consume. De esta forma, se podrá determinar el consumo diario del nodo y así ver la opción que permita mayor duración.

Tabla 4-1 Consumo Nodo "Dispositivo Final"

Dispositivo	Consumo (mA)	Tiempo diario (Horas)	Consumo diario (mAh)
Sensores	50	24	1200
XBee Enviando	40	0.12	4,8
XBee Dormido	0.001	23.88	0.024
Arduino	50	24	1200

Por lo tanto, de la Tabla 4-1 se puede concluir que el consumo diario del nodo "Dispositivo Final" completo es de 2404,82 mAh, por lo que incluso una batería de algunas decenas de Ah se agotará en poco tiempo, por lo que se considera utilizar la energía del ambiente. La opción más conocida es a través de celdas solares, las cuales se adaptan muy bien a este proyecto debido a que el nodo está todo el día recibiendo la radiación del sol. El panel solar se puede conectar directamente al Arduino o por medio de una batería recargable, donde la forma de carga sea a través del panel solar. La última opción es la que se considerará debido a que es importante tener algún respaldo de energía. Es importante tener en cuenta los requerimientos del nodo, debido a que, si se conecta una batería que entrega menos potencia de lo que requiere, incluso el módulo XBee tendrá menos alcance.

4.3.2 Coordinador

El nodo "Coordinador" es el encargado de recibir y organizar la información que recibe desde los nodos "Dispositivo final", además, es el que permite la conexión con internet a través de un servidor local, permitiendo la realización del monitoreo. Este nodo se compone de Raspberry Pi, que será el encargado de organizar la información y realizar el monitoreo; y XBee, que permite la recepción de la información para luego ser procesada.

También se debe considerar la forma de visualizar los datos, puede ser en un monitor conectado por HDMI a la Raspberry Pi o de forma remota en un computador.

Para el desarrollo del proyecto, la visualización se realizó de forma remota, debido a que es más cómodo para la realización completa del proyecto. Sin embargo, por precio y facilidad, se considera mejor opción el monitor.

En la Figura 4-7 se observa la conexión de los dispositivos que se utilizan en este nodo, si bien, se muestra que van todos a la misma fuente de alimentación, la pantalla se conecta directamente a un enchufe.

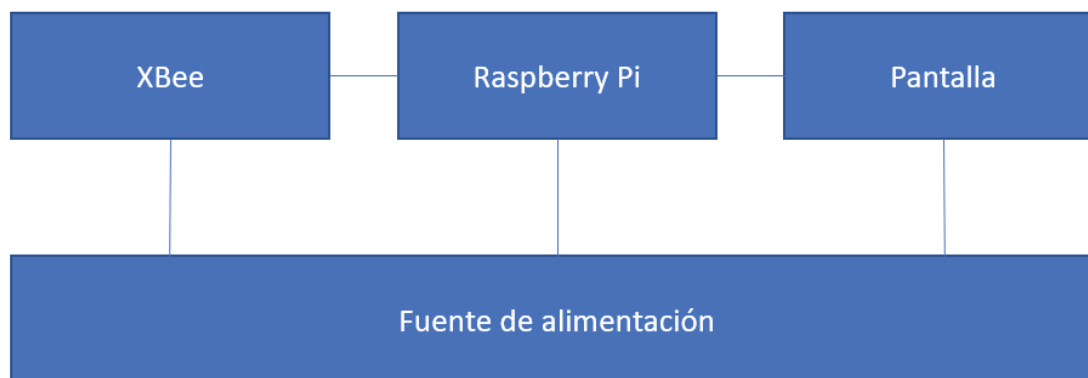


Figura 4-7 Conexión de componentes de Nodo "Coordinador"

4.3.3 Diseño

Para determinar la cantidad de nodos "Dispositivo final" necesarios para obtener información completa del invernadero, se realizaron mediciones en un invernadero estándar (7.5x30m). Un nodo "Dispositivo final" se fue moviendo a través del invernadero cada cierto 10 minutos, con la intención de determinar dónde se obtienen cambios significativos de los datos. El valor que se consideró para esta prueba fue la temperatura, ya que es la más importante al interior de un invernadero.

Se llegó a la conclusión que para determinar la cantidad de nodos "Dispositivo Final" hay que evaluar cada invernadero, ya que se tienen características y ubicaciones distintas, lo cual influye considerablemente. En el invernadero que se realizaron las mediciones se observó que al acercarse a las paredes la temperatura aumenta considerablemente, mientras que al ir recorriendo hacia el centro la temperatura se mantiene relativamente constante. También se pusieron los nodos a lo ancho del invernadero, de esta forma se comprobó que la temperatura no varía significativamente, por lo que no sería necesario poner dos nodos a lo ancho, sino que basta con dejar uno al medio. Con esta información se puede decir que dentro de un invernadero estándar son necesarios cuatro nodos "Dispositivo Final", dos a cada lado cercano a las paredes y los otros dos para captar bien los datos del centro del invernadero. De esta forma, el diseño final del proyecto quedaría como se muestra en la Figura 4-8:

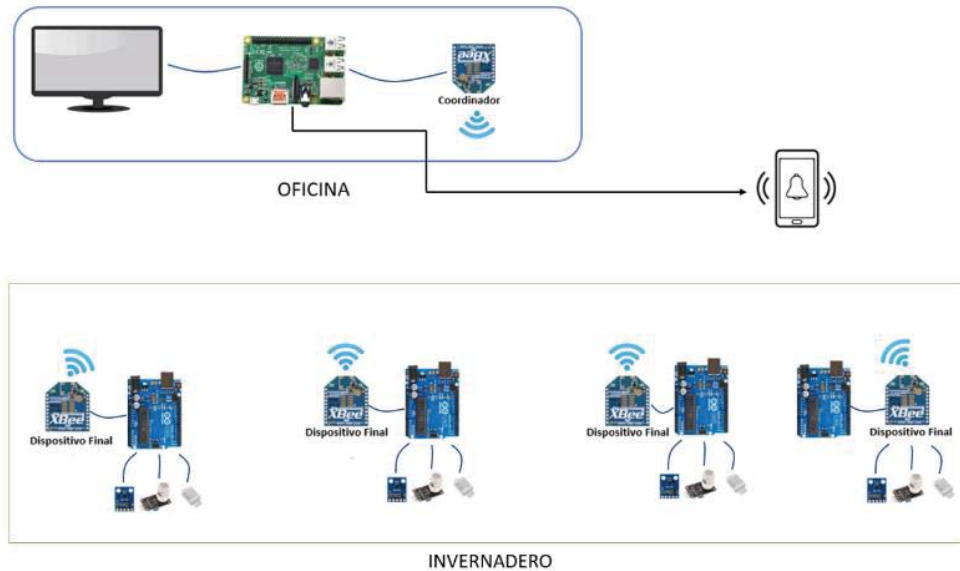


Figura 4-8 Diagrama Final

A pesar de tener el diseño final con cuatro nodos “Dispositivo Final”, para el desarrollo de este proyecto se utilizaron sólo dos nodos, sin embargo, agregar dos más no tiene grandes cambios en el desarrollo.

4.3.4 Diseño de Hardware

Para la realización de este proyecto se utiliza una serie de componentes como lo son los sensores, Arduino, Raspberry Pi y XBee. Todos estos se deben conectar para lograr el monitoreo. A continuación, se mostrará cómo se conecta cada sensor con el Arduino y los módulos XBee con Arduino y Raspberry Pi. En la Figura 4-9, 4-10 y 4-11 se muestran las conexiones del Arduino a los sensores utilizados, mientras que en la Figura 4-12, 4-13 y 4-14 se muestran las conexiones de los XBee al Arduino y Raspberry Pi.

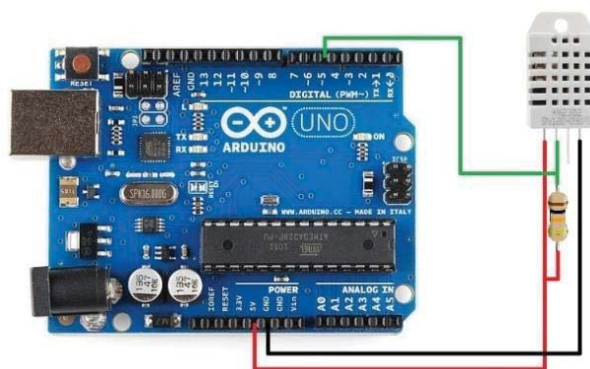


Figura 4-9 Conexión Arduino + DHT22

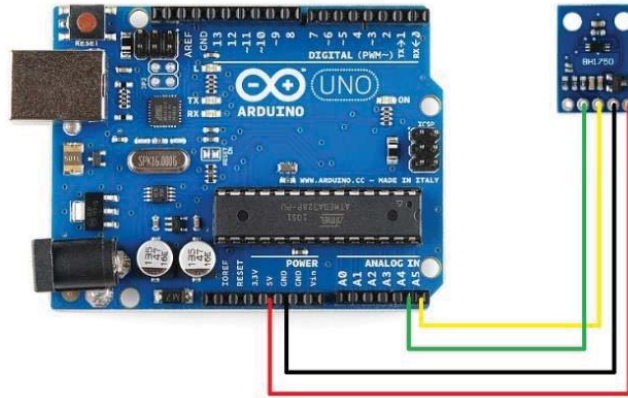


Figura 4-10 Conexión Arduino + BH1750

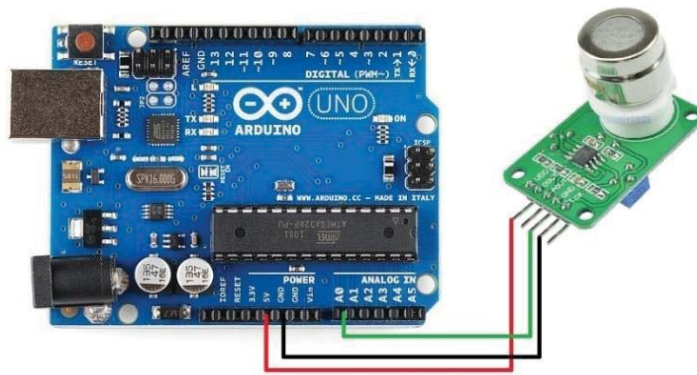


Figura 4-11 Conexión Arduino + MG811

En la conexión de Arduino con XBee se muestra el conexionado por cables (Figura 4-12), sin embargo, esta conexión se puede hacer por medio de una Shield XBee para Arduino (Figura 4-13), de esta última forma es como se hace la conexión para el proyecto.

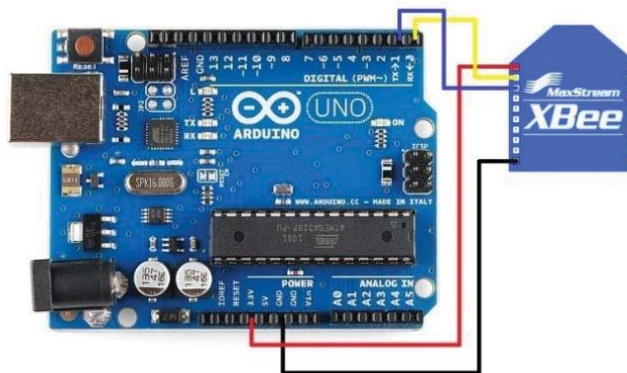


Figura 4-12 Conexión Arduino + Xbee

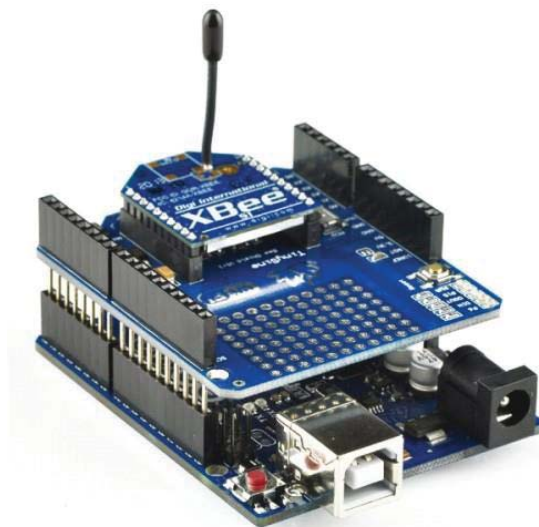


Figura 4-13 Conexión Arduino + Xbee con Shield

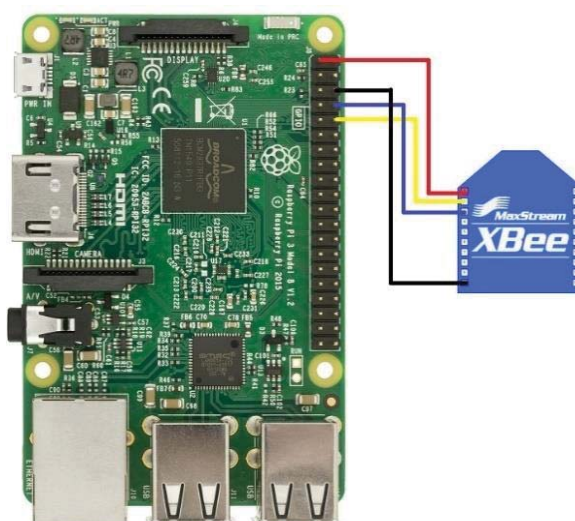


Figura 4-14 Conexión Raspberry Pi + Xbee

4.4 Realización Monitoreo

Para la realización del Sistema de Monitoreo se realizan las conexiones entre los dispositivos, ya sea cableado o inalámbrico, de la forma que se muestra en el apartado anterior. Además, se debió realizar la programación en los nodos “Dispositivo Final” y en el nodo “Coordinador”. En el nodo “Coordinador” se debe realizar la programación necesaria para recibir y separar los datos para después enviar estos datos a una base de datos, luego, se mostrarán los datos en una interfaz gráfica.

4.4.1 Comunicación entre dispositivos

El Arduino será el encargado de recibir la información desde los sensores, por lo que se debe realizar la programación necesaria para recibir los datos de forma correcta. Luego, el Arduino envía los datos obtenidos a la Raspberry Pi a través de los módulos XBee, para lograr esto, se debe activar el envío en Arduino y realizar la programación necesaria en la Raspberry Pi. Cabe destacar que los datos son enviados en modo API, por lo que se debe utilizar una librería en Python para poder trabajar con el mensaje recibido en la Raspberry Pi. A continuación, se muestra el diagrama que explica la programación realizada en los nodos “Dispositivo Final”, donde se obtienen los datos de cada sensor y activa el envío a través del módulo XBee:

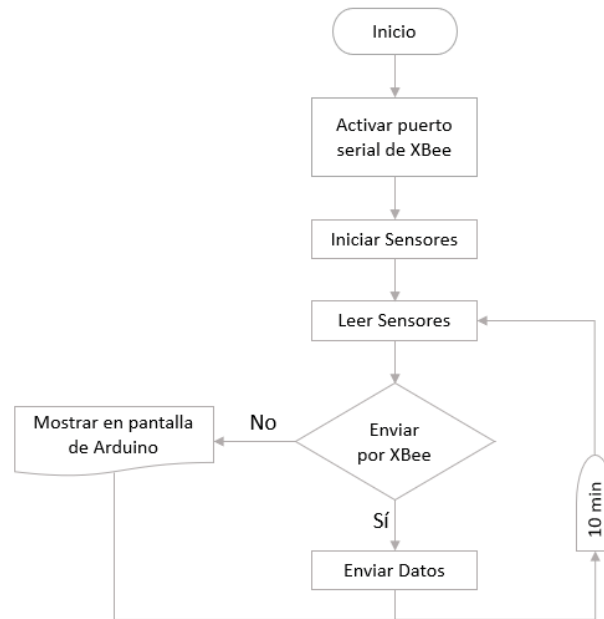


Figura 4-15 Diagrama obtención de datos desde sensores

En el diagrama de la Figura 4-15 se observa que se debe activar el puerto serial para el XBee, de esta forma se podrá realizar el envío de la información. Además, se deben iniciar los sensores, tanto las librerías necesarias para cada sensor como la definición de los pines y variables a utilizar, así se podrán leer los datos y transformarlos adecuadamente para tener los datos en las unidades que se requieren. Una vez obtenidos y transformados todos los valores, se pueden imprimir de forma ordenada en pantalla de Arduino o enviarlo directamente al nodo “Coordinador”.

En las pantallas se mostrará el Nodo desde donde se envía (1 o 2), humedad en porcentaje, temperatura en °C, luminosidad en Lux, CO2 en PPM y voltaje de salida del sensor de CO2. Este último valor es importante para realizar la calibración del sensor, ya que en el código se debe utilizar este valor, el cuál es diferente para cada sensor. En la Figura 4-16 se puede ver la forma en que se imprimen los datos en la pantalla de Arduino, los cuales pueden ser enviados directamente por XBee al nodo “Coordinador”.


```

COM16
[[2-65.40-14.50-14-1041-0.60]]
[[2-65.40-14.50-16-534-0.79]]
[[2-65.60-14.60-13-464-0.83]]
[[2-66.00-14.50-18-433-0.84]]
[[2-65.00-14.60-30-418-0.85]]
[[2-64.60-14.50-10-389-0.87]]
[[2-68.50-14.80-10-396-0.87]]
[[2-69.30-15.10-10-403-0.86]]
[[2-67.40-15.20-10-396-0.87]]
[[2-65.30-15.30-9-396-0.87]]

```

Figura 4-16 Resultados nodo "Dispositivo Final"

Una vez realizada la programación del nodo “Dispositivo Final”, se debe realizar la programación para recibir la información en el nodo “Coordinador”. Como se mencionó antes, la información que recibe el módulo XBee se obtiene en modo API, esto implica que, además de los datos enviados por los nodos “Dispositivo Final”, envía información extra de cada módulo XBee que envía, como lo es la dirección de cada módulo. Es por esto que se utiliza una librería para la separación de la información recibida [39]. A continuación, se muestra el diagrama de la programación en Python necesaria para recibir la información y separarla.

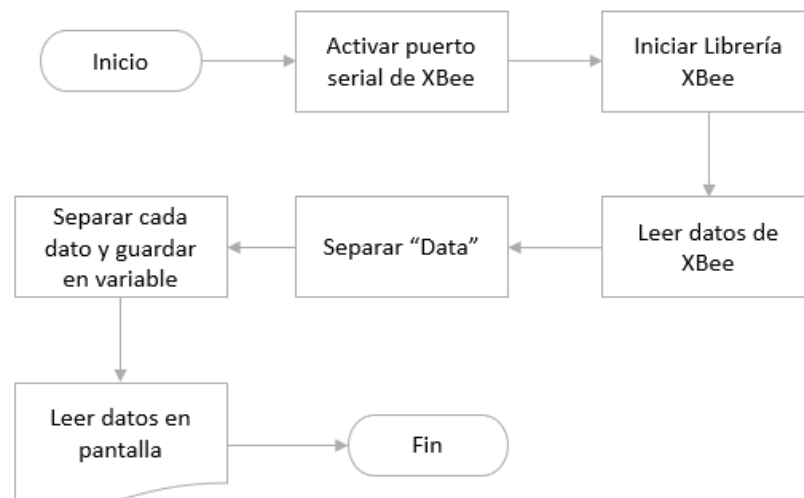
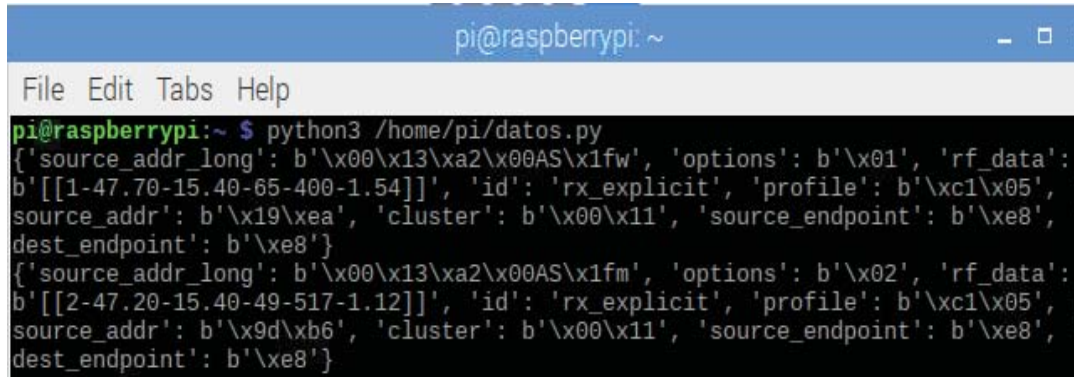


Figura 4-17 Diagrama recepción y organización de datos

En el diagrama de la Figura 4-17 se observa que se inició la librería XBee, esta permitirá separar la información recibida, además, se debe activar el puerto que está siendo utilizado por el módulo XBee para recibir la información. Luego se leerá la información que está siendo recibida para luego separarla mediante la librería XBee. Finalmente se selecciona solo la información que corresponde a los datos que se enviaron desde los nodos “Dispositivo Final” y se separan los datos de cada sensor y se guarda en variables, para luego poder leer los datos separados en pantalla.

Al mostrar en pantalla la obtención de la información completa recibida por la librería XBee, se observará que muestra distintos datos, como lo es la dirección de cada módulo del que recibe la información, la información recibida, la longitud de la dirección, entre otros. De estos, el que importa es el de los datos recibidos 'rf_data'. En la Figura 4-18 y Figura 4-19 Se observa la información recibida por la librería XBee y los datos obtenidos de 'rf_data' respectivamente.

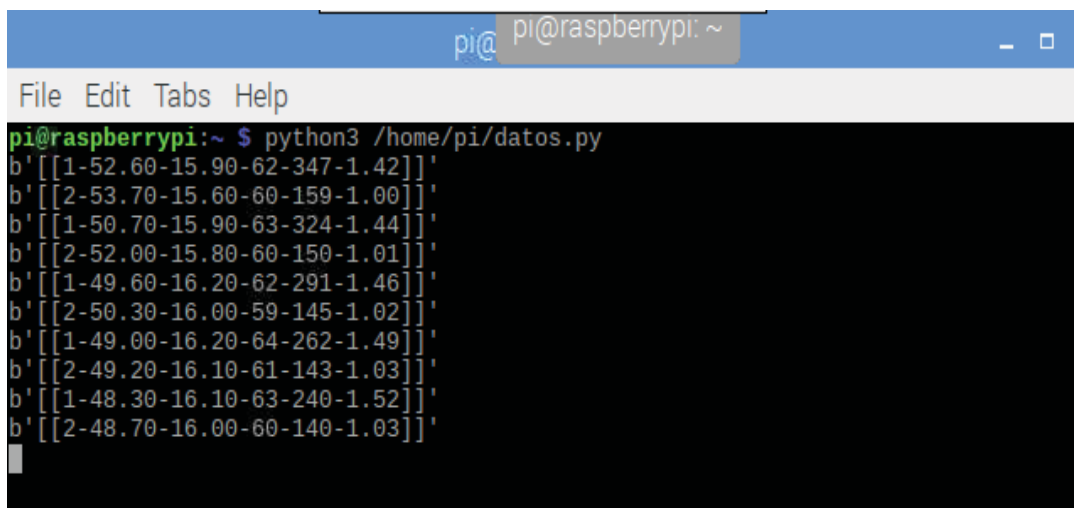


```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ python3 /home/pi/datos.py
{'source_addr_long': b'\x00\x13\xa2\x00AS\x1fw', 'options': b'\x01', 'rf_data':
b'[[1-47.70-15.40-65-400-1.54]]', 'id': 'rx_explicit', 'profile': b'\xc1\x05',
source_addr': b'\x19\xea', 'cluster': b'\x00\x11', 'source_endpoint': b'\xe8',
dest_endpoint': b'\xe8'}
{'source_addr_long': b'\x00\x13\xa2\x00AS\x1fm', 'options': b'\x02', 'rf_data':
b'[[2-47.20-15.40-49-517-1.12]]', 'id': 'rx_explicit', 'profile': b'\xc1\x05',
source_addr': b'\x9d\xb6', 'cluster': b'\x00\x11', 'source_endpoint': b'\xe8',
dest_endpoint': b'\xe8'}

```

Figura 4-18 Obtención de información de librería ZigBee



```

pi@raspberrypi: ~
File Edit Tabs Help
pi@raspberrypi:~ $ python3 /home/pi/datos.py
b'[[1-52.60-15.90-62-347-1.42]]'
b'[[2-53.70-15.60-60-159-1.00]]'
b'[[1-50.70-15.90-63-324-1.44]]'
b'[[2-52.00-15.80-60-150-1.01]]'
b'[[1-49.60-16.20-62-291-1.46]]'
b'[[2-50.30-16.00-59-145-1.02]]'
b'[[1-49.00-16.20-64-262-1.49]]'
b'[[2-49.20-16.10-61-143-1.03]]'
b'[[1-48.30-16.10-63-240-1.52]]'
b'[[2-48.70-16.00-60-140-1.03]]'

```

Figura 4-19 Obtención de datos 'rf_data'

4.4.2 Base de Datos

Para el desarrollo de la base de datos se requiere de un conjunto de sistemas, lenguajes y tecnologías, donde cada uno realiza una función específica y en conjunto permitirán la obtención y visualización de la base de datos. Existe un conjunto muy conocido que permitirán realizar una base de datos, la cual se denomina LAMP (Figura 4-20), este es el conjunto de las siguientes tecnologías: Linux, que es el sistema operativo a utilizar, en el caso de la Raspberry Pi, utiliza Raspbian; Apache, el cual es el servidor y permitirá la conexión y posterior visualización de la base de datos mediante una página web; MySQL, que realiza la gestión de la base de datos; Python, el cual es el lenguaje de programación a utilizar, se realiza el código en Python para el envío de los datos [40]. Además, se utiliza phpMyAdmin para visualizar la base de datos en una tabla.

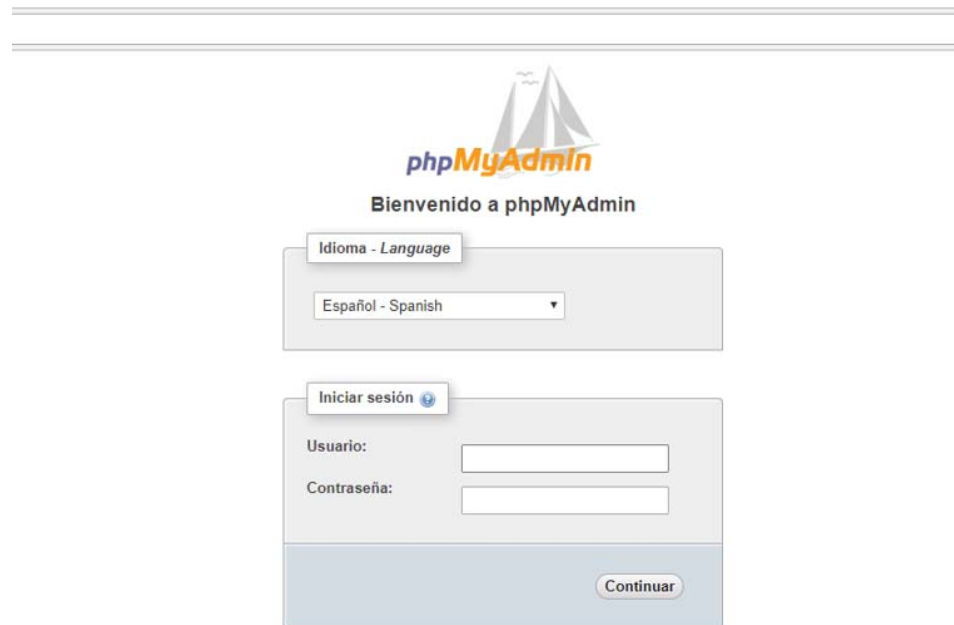


Figura 4-22 Inicio phpMyAdmin

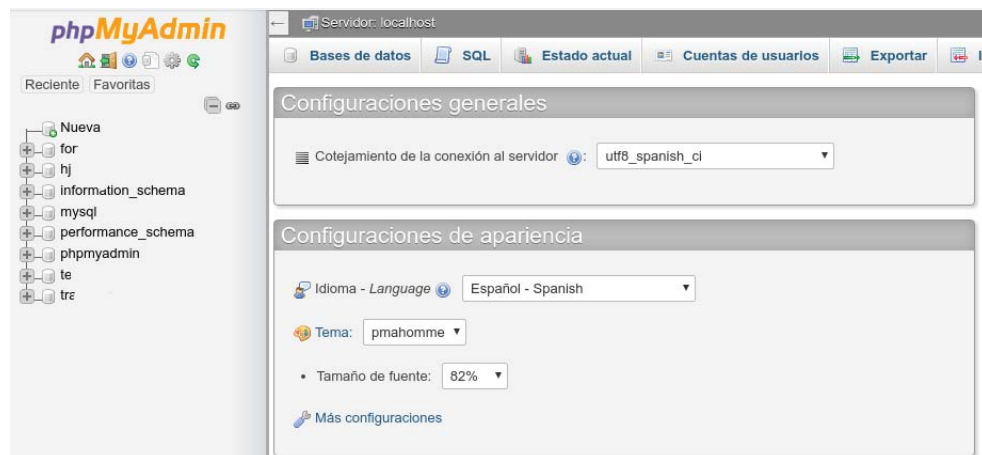


Figura 4-23 Acceso a phpMyAdmin

En la Figura 4-21 se observa que se puede acceder con éxito desde el computador al servidor creado en la Raspberry Pi. La Figura 4-22 muestra el inicio de phpMyAdmin, donde pide la el usuario y la contraseña que se debe establecer en la instalación. Finalmente, en la Figura 4-23 se puede observar dónde se pueden realizar las bases de datos para luego poder realizar el monitoreo de las variables. Cabe destacar que la base de datos se puede crear directamente desde Python, así, al recibir los datos y separarlos, se subirán directamente a la base de datos.

Una vez instalado Apache, MySQL y phpMyAdmin, se realiza el código en Python que permitirá el envío de los datos a la base de datos, la cual se visualizará al ingresar a phpMyAdmin. En la Figura 4-24, se muestra el diagrama de la programación que permite la creación de la base de datos [42].

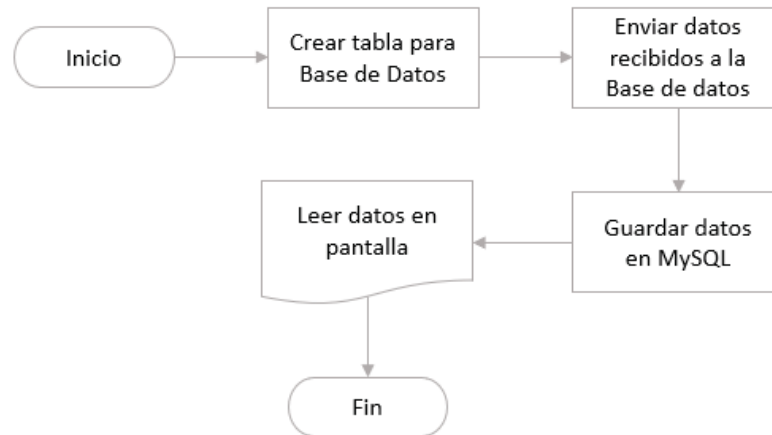


Figura 4-24 Diagrama de flujo creación Base de Datos

Para crear la tabla para la base de datos y acceder a ella se requieren algunos datos como lo son el host, usuario, contraseña y nombre de la base de datos. Estos datos son los que se utilizan en phpMyAdmin, donde usuario y contraseña son las utilizadas para ingresar y monitoreo es el nombre de la base de datos que se podrá visualizar. En la Figura 4-25 se observa la base de datos que se puede visualizar en phpMyAdmin, donde se va llenando a medida que el nodo “Coordinador” recibe los datos.

			id_datos	router	fecha	hora	temperatura	humedad	luminosidad	co2
<input type="checkbox"/>			1	1	2018-06-21	22:29:10	15.8	66.8	24	53
<input type="checkbox"/>			2	2	2018-06-21	22:29:16	15.1	49.8	17	22
<input type="checkbox"/>			3	1	2018-06-21	22:29:32	15.7	60.4	24	51
<input type="checkbox"/>			4	2	2018-06-21	22:29:38	15.1	49.6	18	22
<input type="checkbox"/>			5	1	2018-06-21	22:29:55	15.8	56.4	24	50
<input type="checkbox"/>			6	2	2018-06-21	22:30:00	15.1	49.5	18	22
<input type="checkbox"/>			7	1	2018-06-21	22:30:17	15.8	53.1	23	50
<input type="checkbox"/>			8	2	2018-06-21	22:30:23	15.1	49.5	14	22
<input type="checkbox"/>			9	1	2018-06-21	22:30:39	15.8	51.1	23	51
<input type="checkbox"/>			10	2	2018-06-21	22:30:45	15.1	49.6	14	22

Figura 4-25 Base de datos en phpMyAdmin

4.4.3 Interfaz gráfica

Una vez realizada la base de datos, se puede trabajar con ella y realizar la interfaz gráfica, donde se mostrarán los gráficos de temperatura, humedad, luminosidad y CO2, además la opción de ver los datos actuales, una fecha específica, los últimos 7 días o el histórico y la opción de exportar los datos. A continuación, se muestra el diagrama de la programación en PHP necesaria para la realización de la Interfaz Gráfica [43].

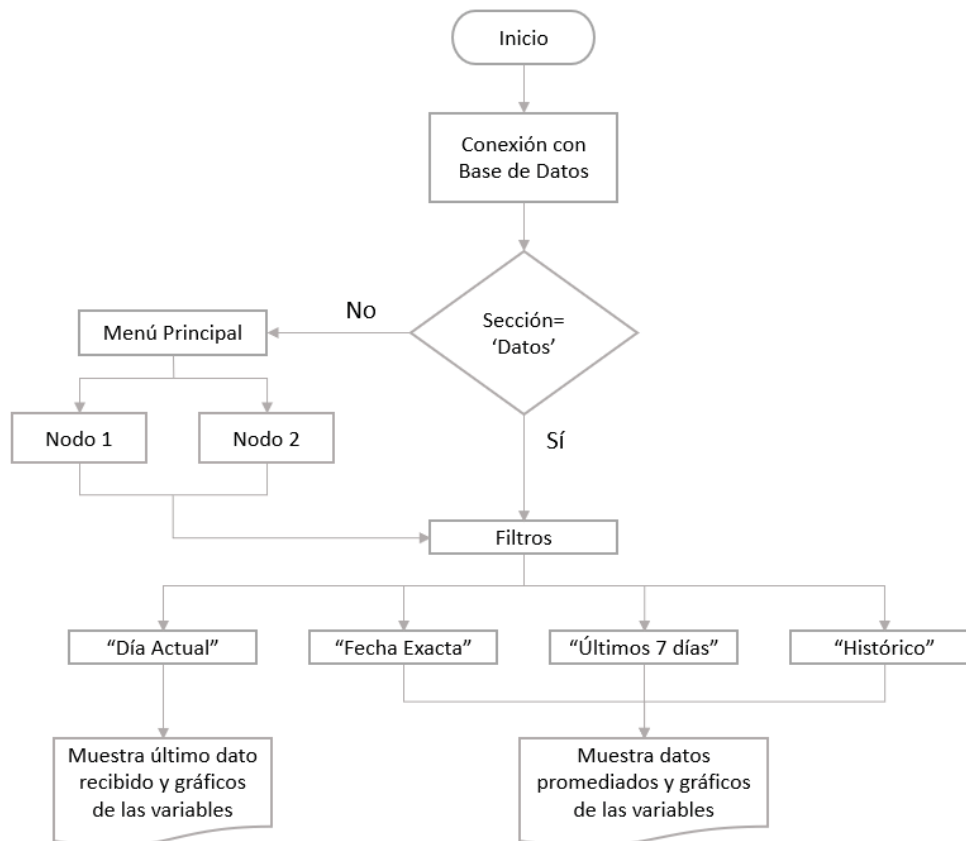


Figura 4-26 Diagrama de flujo Interfaz Gráfica

En el diagrama de la Figura 4-26 se observa que lo primero que se debe hacer es realizar la conexión con la Base de Datos creada anteriormente para poder visualizar la información de esta. Luego se tiene la opción del menú principal o que esté mostrando los gráficos e información. En caso de acceder al menú principal, se tienen dos opciones, acceder a la información del "Nodo 1" o a la del "Nodo 2", donde se tendrán distintos filtros dependiendo los datos que se quieran visualizar. Se tienen las opciones de "Día Actual", donde se mostrarán los gráficos de las variables, donde se verán todos los datos del día en que se están visualizando los datos y, además, se verá en pantalla el último dato recibido de cada variable. Los otros filtros posibles son "Fecha Exacta", donde se verán los datos del día que se elija, "Últimos 7 días", que mostrará los datos de la última semana desde que se está visualizando la interfaz, "Histórico", que muestra todos los datos de la base de datos. Estos últimos tres filtros mostrarán en pantalla los datos promediados y los gráficos correspondientes a cada uno. Además, se tiene la opción de exportar los datos desde la Interfaz.

A continuación, se muestra la Interfaz Gráfica obtenida, donde se tienen dos pantallas, una que muestra el menú que permite la selección de cada nodo (Figura 4-27) y la otra pantalla que muestra los datos con los filtros realizados (Figura 4-28).



Figura 4-27 Menú principal Interfaz Gráfica

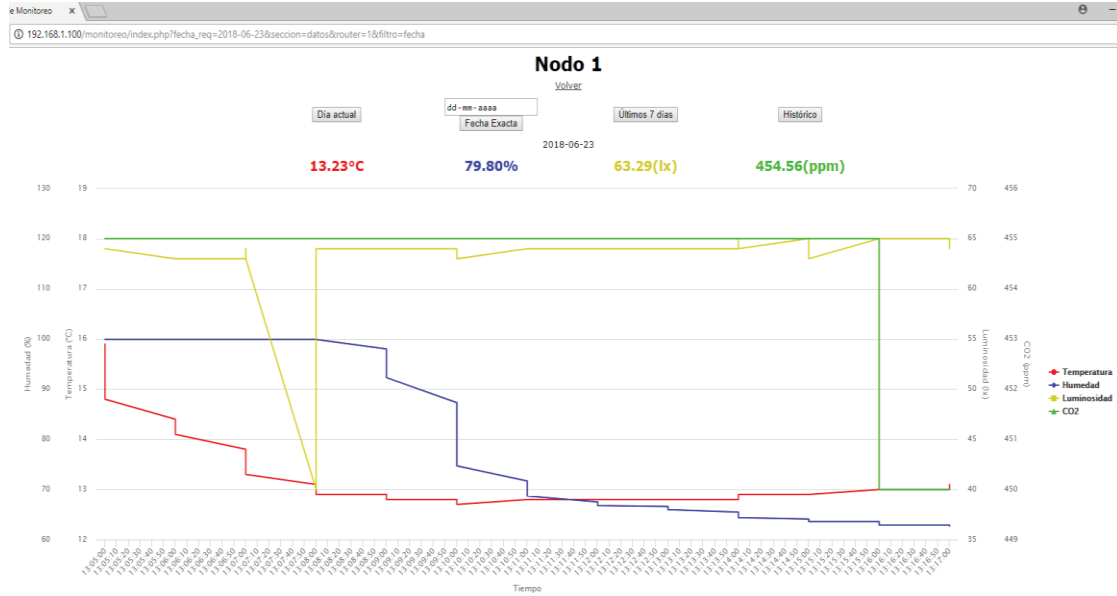


Figura 4-28 Interfaz Gráfica

4.5 Realización Alarma

Para la realización del Sistema de Alarma se utiliza el mismo Hardware que el Sistema de Monitoreo, el cambio se realiza una vez que se obtienen los datos, ya que se debe realizar la programación necesaria para trabajar con los datos con el fin de enviar correctamente la alarma cuando sea necesario. Para esto se debe determinar el modo de envío y los límites inferiores y superiores de cada variable para así tener una alarma con valores estándar. Si bien, los límites dependerán del requerimiento de cada invernadero, se tienen valores de los límites teóricos para poder estandarizar la alarma.

4.5.1 Envío por E-mail

Para el envío por E-mail no se requiere de algún dispositivo extra, pero es necesario tener internet. Para la realización de esta alarma se debe realizar la programación necesaria para trabajar con los datos obtenidos de la forma en que se mostró en el Sistema de Monitoreo. En la Figura 4-29, se muestra el diagrama de la programación realizada para lograr el envío de la alarma vía E-mail [44].

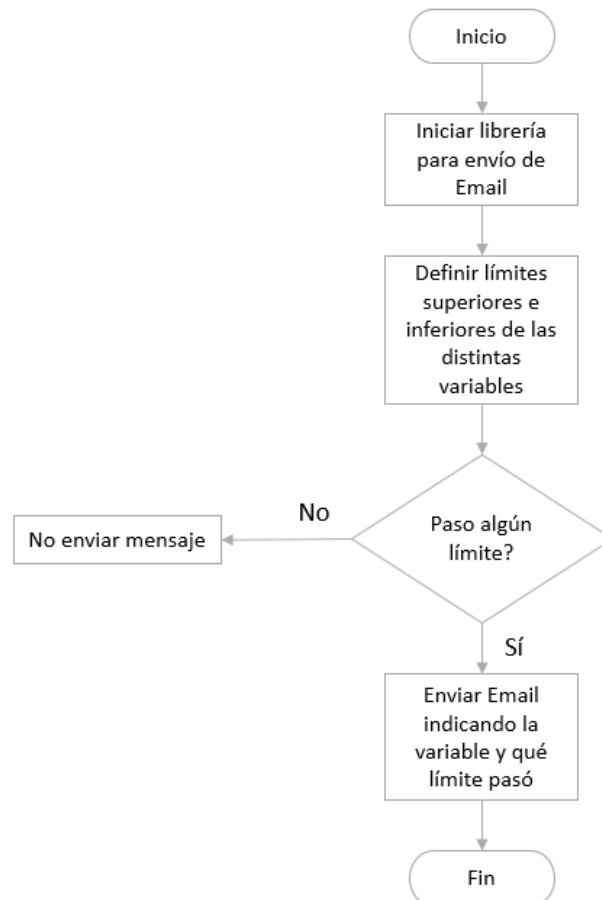


Figura 4-29 Diagrama de flujo Alarma

Una vez que se reciben los datos en el Nodo “Coordinador” desde los Nodos “Dispositivo Final”, se trabajará con ellos para lograr el envío del correo electrónico. Primero se debe instalar la librería de Python `smtplib`, la cual permitirá el envío. Luego se crea la función que permitirá realizar el envío cada vez que sea llamada, donde se prepara el envío, se realiza la conexión con el servidor SMTP, inicia la sesión y se envía el correo electrónico. Además, se deben definir los límites de cada variable, así, si sobrepasa el límite superior dirá, por ejemplo, “Temperatura alta” y, si sobrepasa el límite inferior, el mensaje será “Temperatura baja”. Una vez determinados los límites puede comenzar a funcionar la alarma, de modo que, si el valor medido pasa alguno de los límites determinados, se enviará el E-mail indicando la variable que está fuera de sus límites y qué límite pasó, por otra parte, si el valor medido no pasa ningún límite, no envía nada.

4.5.2 Límites

Existen límites estándar para las variables que se van a medir, estos límites son establecidos porque sobre o bajo los valores se sufren daños en las plantas o se detiene su crecimiento, por lo que manteniendo las variables dentro de los límites permiten un crecimiento óptimo. En la Tabla 4-2 se muestran los valores máximos y mínimos que debería tener un invernadero de paltos y pimientos. Se estudian los paltos y pimientos porque las pruebas que se realizan en terreno son en invernaderos de palto y pimientos. [45][8]

Tabla 4-2 Límites de variables en invernaderos

Variable	Mínimo Palto	Máximo Palto	Mínimo Pimiento	Máximo Pimiento
Temperatura	15 °C	25 °C	18 °C	28 °C
Humedad	60 %	70 %	50 %	70%
Luminosidad	200 lux	15000 lux	3000 lux	20000 lux
CO ₂	200 ppm	800 ppm	200 ppm	500 ppm

A pesar de tener valores máximos y mínimos estándar de las variables, estos límites dependerán de cada cliente, ya que dependerá de la ubicación del invernadero y las condiciones que se logren tener al interior.

Debido a que se tienen distintas condiciones en cada invernadero, antes de instalar definitivamente el sistema a cada cliente, se tendrá que realizar una calibración con el fin de analizar cada invernadero y así dejar los límites más adecuados para cada caso, de esta forma tener los valores óptimos de cada cultivo.

4.6 Diseño Carcasa 3D

Es importante que los dispositivos utilizados estén protegidos de las partículas o algún imprevisto, por ejemplo, en un invernadero es común que, al tener cambios de temperatura, caiga agua del techo y eventualmente podría caer directamente en un nodo. Además, permite que las conexiones realizadas queden estables. Es por esto que se diseñó una carcasa para cada nodo, la cual será impresa en una impresora 3D.

En la Figura 4-30 se muestra el diseño de la carcasa para el nodo "Dispositivo final", que contiene Arduino, módulo XBee y los sensores. Esta carcasa tiene medidas de 11 x 8.5 x 6 cm, además, tiene agujeros para los distintos sensores y para el módulo XBee. Para el sensor DHT22 se tiene un agujero rectangular de 1.8 x 2.3 cm, para el sensor MG-811 un agujero circular de 2 cm de diámetro, para el sensor BH-1750 un agujero rectangular de 1.7 x 0.9 cm y para el módulo XBee un agujero rectangular de 5.85 x 1 cm. Estos agujeros se encuentran a los costados de la carcasa, a excepción del sensor BH-1750 que se encuentra en la tapa para que capte de la mejor forma posible la luminosidad.

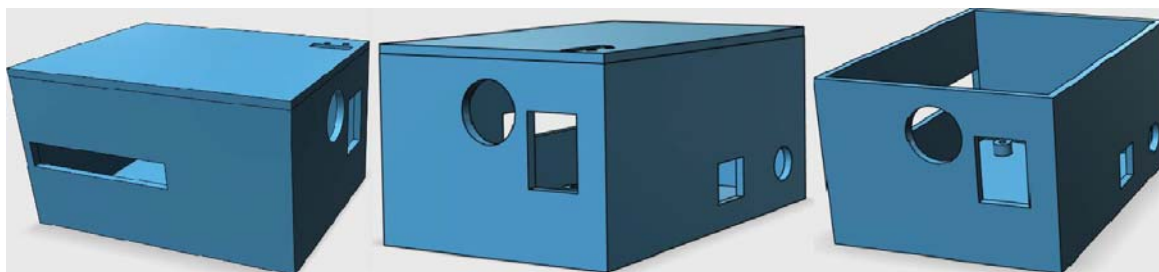


Figura 4-30 Diseño Nodo "Dispositivo Final"

En la Figura 4-31 se muestra el diseño de la carcasa para el nodo "Coordinador", que contiene la Raspberry Pi y el módulo XBee. En esta carcasa se tienen medidas diferentes para la parte de la Raspberry Pi y del módulo XBee, para la Raspberry Pi mide 8.7 x 6 x 2.5 cm y para el módulo XBee mide 7.5 x 5.5 x 3.8 cm, además, se agregó un espacio para los cables y así realizar las conexiones sin problemas, este mide 5 x 2 x 2 cm. También tiene los agujeros necesarios para la Raspberry Pi y otro para la antena del módulo XBee.

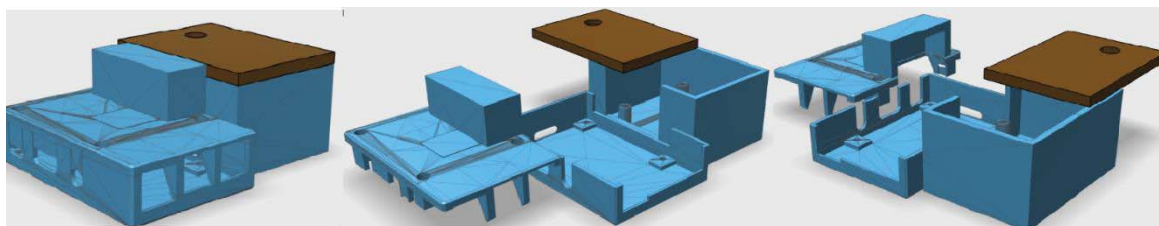


Figura 4-31 Diseño nodo "Coordinador"

De esta forma, el diseño impreso, con sus respectivos dispositivos queda como se muestra a continuación, donde en la Figura 4-32 se puede observar el Nodo “Dispositivo Final” y en la Figura 4-33 el Nodo “Coordinador”.



Figura 4-32 Nodo "Dispositivo Final"



Figura 4-33 Nodo "Coordinador"

4.7 Pasos a seguir para instalación

Para llegar a implementar este sistema de monitoreo y alarma en un invernadero es necesario realizar una serie de acciones que permitan el funcionamiento óptimo de este. Como se ha mencionado, cada invernadero tendrá distintas condiciones debido a que se pueden encontrar en distintos lugares del país y Chile tiene un clima muy variado, además, también dependerá de cómo esté construido cada invernadero.

Cuando se vaya a instalar el sistema en un invernadero se debe realizar un trabajo previo. Antes de instalar los nodos “Dispositivo Final” al interior del invernadero se debe realizar una evaluación para determinar la cantidad de nodos que requerirá este, esto dependerá de la homogeneidad del ambiente, la altura del invernadero, si tiene muchas fugas. Se realizan pruebas con un nodo que se irá moviendo al interior del invernadero, de forma de evaluar cuánto van cambiando los valores, así se podrá determinar la cantidad que requerirá el invernadero.

Una vez determinada la cantidad de nodos que se deben instalar en el invernadero, se deben calibrar los sensores, de forma que entreguen valores lo más certeros posible y que representen adecuadamente los valores que tiene el invernadero.

También hay que corroborar que la distancia entre los nodos “Dispositivo Final” al interior del invernadero con el nodo “Coordinador” se encuentra dentro del alcance de los módulos XBee, de forma de asegurar que los datos llegarán en todo momento y no se tendrá pérdida de información.

Además, se deben determinar los valores que tendrán los límites máximos y mínimos de las variables. Estos valores dependerán de los requerimientos del cliente y de las condiciones que se logran en cada invernadero. Hay que tener en cuenta que se deberá determinar los límites para cada temporada, donde las más notorias son invierno y verano, pero podría ser otra temporada dependiendo el cultivo que se tenga. Es importante determinar estos valores debido a que se tendrán valores muy distintos dependiendo las temporadas, por ejemplo, de invierno a verano, la temperatura puede variar en aproximadamente 10°C. Otro factor importante a considerar al momento de determinar los límites es el lugar donde se encuentra, ya que puede encontrarse en el norte, centro o sur del país y las variaciones entre estas zonas serán considerables.

Una vez determinados y programados los valores máximos y mínimos de las variables se pueden instalar los nodos ya calibrados en los lugares estratégicos para que permita obtener valores representativos del lugar. Generalmente, los invernaderos no cuentan con alguna estructura para instalar los nodos, por lo que hay que considerar la realización de la estructura que permita dejar los nodos a la altura necesaria.

5 Resultados

Para corroborar el funcionamiento del Sistema de Monitoreo se realizaron pruebas en una habitación, realizando distintas acciones para que los valores de los sensores cambien. Además, se realizaron pruebas en terreno, donde se fue a dos invernaderos distintos; uno ubicado en Limache y que produce pimientos; el otro ubicado en la Escuela de Agronomía de la PUCV en Quillota, el cual produce paltos.

5.1 Pruebas sensores

Antes de ver los resultados de las pruebas realizadas en terreno es necesario ver las diferencias que tienen los valores del Nodo 1 con el Nodo 2, de esta forma se podrán sacar conclusiones más certeras. Al poner ambos Nodos en el mismo lugar, los valores de temperatura y humedad deberían ser los mismos, mientras que los valores de luminosidad y CO₂ pueden llegar a ser muy diferentes dependiendo de la posición o la calibración en el caso del CO₂. En la Figura 5-1 se mostrarán los valores obtenidos de temperatura y humedad de cada Nodo al ubicarlos en el mismo lugar.

```
Nodo: 1 Temperatura: 19.40 Humedad: 60.40  
Nodo: 2 Temperatura: 20.30 Humedad: 60.90
```

Figura 5-1 Temperatura y Humedad ambos Nodos

En la Figura 5-1 se puede observar que hay aproximadamente 1 °C y 0.5% HR de diferencia entre ambos Nodos, lo cual se considerará al momento de ver los resultados.

5.2 Invernadero Limache

Las pruebas en el invernadero de Limache se realizaron el día 27 de junio durante la mañana, donde el día estuvo mayormente soleado, por lo que la temperatura al interior del invernadero es considerablemente alta, sin embargo, los pimientos soportan grandes temperaturas. El invernadero tiene las medidas estándar, 7.5 x 30 metros, con una altura de 2.5 metros, lo cual no es muy alto para un invernadero. Al ser de poca estatura, la masa de aire al interior se concentra fácilmente, lo cual genera cambios más drásticos de las variables.

Antes de comenzar a realizar las mediciones, se debe realizar la calibración del sensor de Dióxido de Carbono, donde se considera que al aire libre hay aproximadamente 0.04% de concentración de CO₂, lo cual equivale a 400 ppm. Se tuvieron algunos problemas con este sensor debido a que es muy sensible al movimiento, sobre todo al mover los cables. Para realizar la calibración se debe cargar el programa de Arduino con un valor actualizado, y al hacer esto ocurre que se mueve un poco y varía el valor, algunas veces considerablemente. Por lo tanto, en esta medición, se puede observar el correcto funcionamiento de este sensor en el Nodo 1, y en el Nodo 2 hubo algunos problemas para calibrar correctamente.

El invernadero cuenta con 5 hileras de pimientos a lo ancho, por lo que los Nodos se pusieron sobre la hilera del medio, a aproximadamente 7.5 metros de cada pared a lo largo, es decir, hay una distancia de 15 metros entre el Nodo 1 y Nodo 2, y a 2 metros de alto. En la Figura 5-2 y 5-3 se puede apreciar la ubicación de los Nodos en el invernadero, así se puede comprender de mejor forma el lugar donde se encuentran los Nodos. Para la toma de datos se pensó en tomarlos cada 15 minutos, de esta forma obtener la mayor cantidad de datos y tener un gráfico más representativo, sin embargo, debido a la calibración que se comentó antes se decidió realizar la toma de datos cada 10 minutos, para así no esperar tanto y obtener la mayor cantidad de datos una vez realizada la calibración.

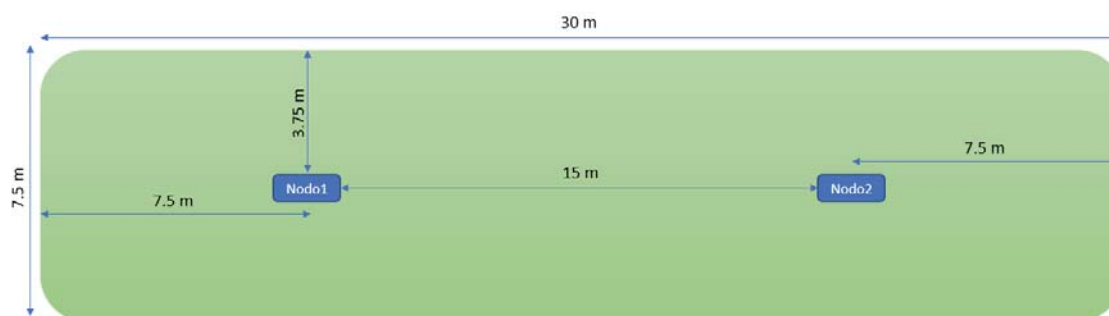


Figura 5-2 Ubicación de Nodos

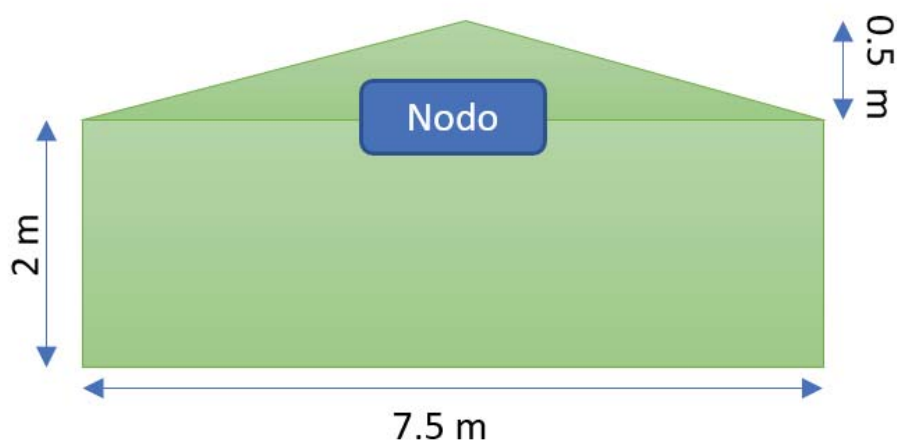


Figura 5-3 Ubicación de Nodos en altura

A continuación, se muestran los resultados de la Interfaz Gráfica obtenida en este invernadero del Nodo 1 y Nodo 2, donde se verá la temperatura y humedad en un gráfico y la luminosidad y CO₂ en otro, esto con el fin de comprender de mejor forma los gráficos.

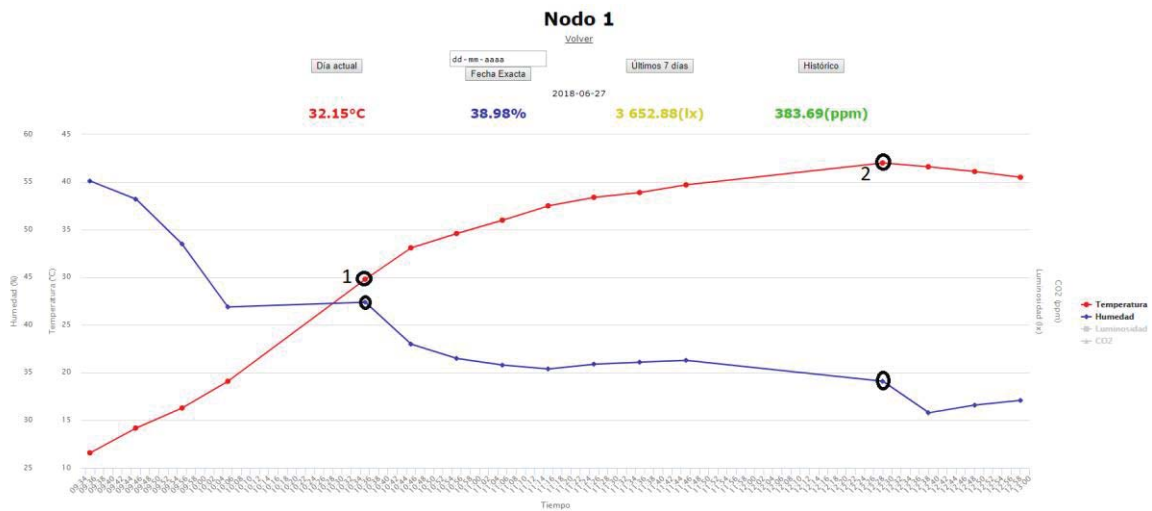


Figura 5-4 Temperatura y Humedad Nodo 1

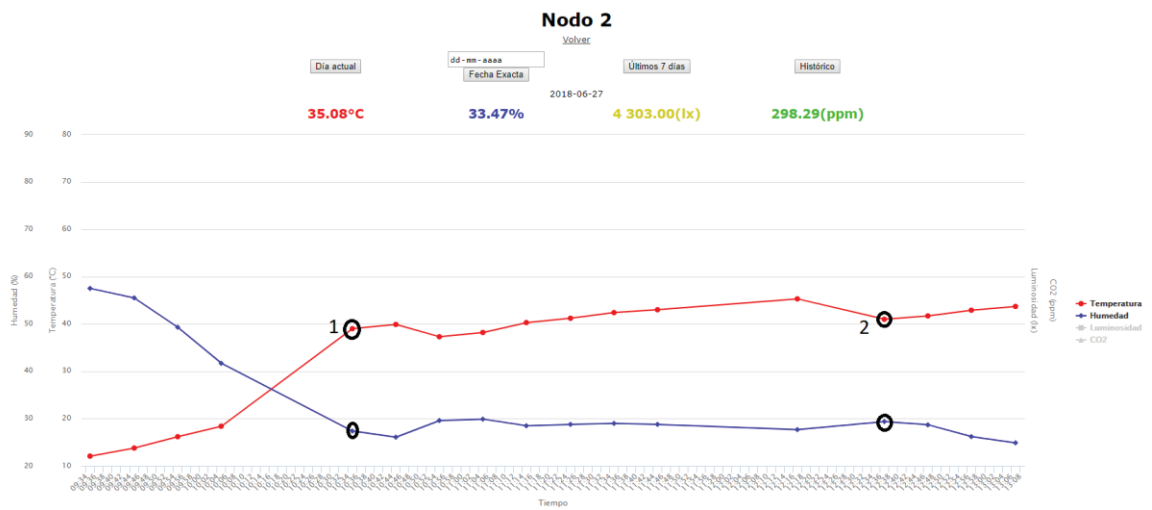


Figura 5-5 Temperatura y Humedad Nodo 2

En la Figura 5-4 y Figura 5-5 se muestra la temperatura en rojo y la humedad en azul y se puede observar que ambos nodos tienen gráficas similares, sin embargo, con una temperatura promedio con una diferencia de aproximadamente 3 °C, donde 1 °C se debe a la diferencia propia de los sensores, los otros 2 °C se deben a la posición de los sensores, ya que, al llegar el sol directamente a los sensores, aumenta más la temperatura, y al llegar de distinta forma a cada Nodo se obtiene esta diferencia. Las primeras mediciones son tomadas al aire libre, para ver la diferencia al ingresar los Nodos al invernadero, en el punto 1 marcado en las gráficas se muestra el primer dato tomado dentro del invernadero, donde se puede observar que la temperatura aumenta y la humedad disminuye considerablemente, lo cual es normal debido a las condiciones de un

invernadero. La temperatura va aumentando a medida que pasa la hora porque el sol va subiendo y cada vez llega más directamente a los sensores. Luego, en el punto 2 de los gráficos, se abren las cortinas del invernadero, por lo que el aire comienza a circular y la temperatura tiende a bajar, sin embargo, en el Nodo 2 no comienza a bajar, lo cual se debe a la posición del sensor y del sol en ese momento; y la humedad también baja debido a que la humedad en el invernadero se debe principalmente a la transpiración de las plantas, y al abrirse las cortinas este efecto disminuye.

Después del punto 2 se comienza a cambiar de posición el nodo 1, sin embargo, no se tenía mucho más tiempo para esperar a que se estabilizara el ambiente luego de abrir las cortinas. Al poner el Nodo 1 en la hilera siguiente de donde se encuentra y a al lado del Nodo 2, los valores son prácticamente iguales. Luego se fue variando la posición del Nodo 1 en la hilera del medio, llegando a la conclusión que en el medio del invernadero la temperatura y humedad no varían considerablemente, ya que no hay más de 1 °C de diferencia, mientras que al ubicar el Nodo cerca de las cortinas los valores cambian bastante, teniendo 3 °C de diferencia con el medio.



Figura 5-6 Luminosidad y CO2 Nodo 1

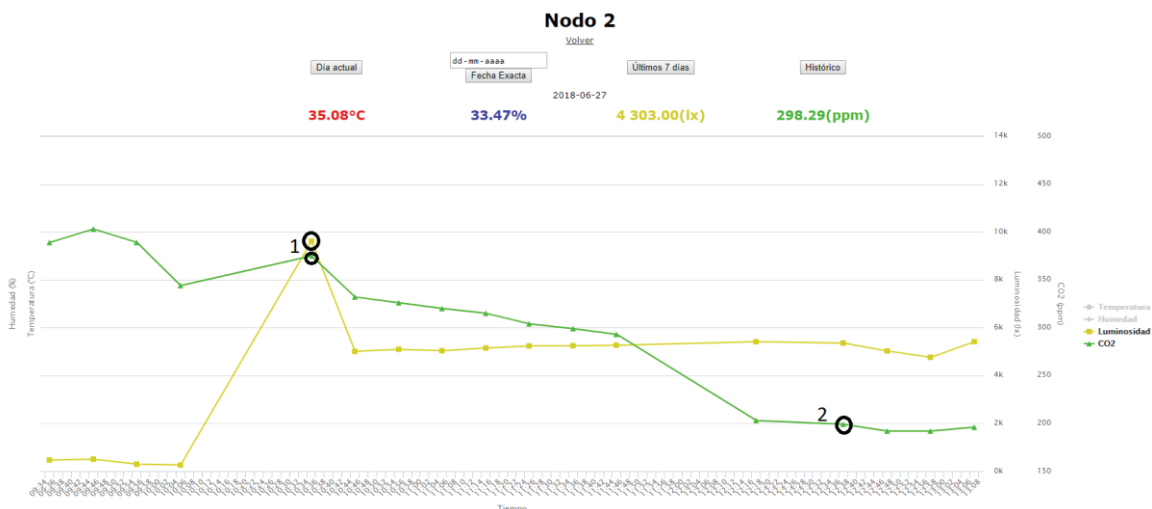


Figura 5-7 Luminosidad y CO₂ Nodo 2

En la Figura 5-6 y 5-7 se muestra la luminosidad en amarillo y CO₂ en verde. Aquí se pueden observar grandes diferencias entre el Nodo 1 y Nodo 2. Las diferencias de la luminosidad se deben a la posición de los sensores y a un error del diseño de la tapa de la carcasa, ya que el agujero que se dejó para este sensor es muy pequeño y dependiendo de la posición del sol se forma una sombra, de esta forma en algún momento no representa correctamente la luminosidad. Las diferencias del CO₂ se debe principalmente a que hubo problemas para calibrar adecuadamente el Nodo 2 y arrojó valores mucho menores, sin embargo, se observa que las reacciones a las acciones que se realizaron son similares en ambos Nodos. En el punto 1 de la gráfica, donde se ingresan los Nodos al invernadero, se observa una baja de CO₂ significativa en comparación con la que se tiene al aire libre, lo cual es normal en la situación que se tiene, el invernadero cerrado toda la noche y antes de abrir las cortinas se tiene una baja de CO₂ y se puede llegar hasta 0,005%. Luego, en el punto 2, donde se abren las cortinas, comienza la circulación del aire y lo normal es que aumente el CO₂.

5.3 Invernadero Quillota

Las pruebas en el invernadero de Quillota se realizaron el día 3 de julio durante la mañana y parte de la tarde, donde el día estuvo nublado y en varias ocasiones se pronunciaba el sol, lo cual generaba un aumento en la temperatura al interior del invernadero. Este invernadero tiene medidas de 8 x 15 metros con una altura de 5 metros, esta altura permite una mayor homogeneidad en el ambiente debido a que se tiene una masa de aire mucho más grande y cuesta que cambien las variables, es decir, los cambios no son tan drásticos como en el invernadero de Limache. Cabe destacar que este invernadero cuenta con un punto donde realizan la toma de datos de temperatura y humedad por medio de un datalogger, el cual realiza mediciones cada 30 minutos y se extraen una vez a la semana. Esto permite realizar la comparación entre el datalogger y los Nodos, de esta forma se podrán realizar más conclusiones.

El invernadero cuenta con 3 hileras de paltos a lo ancho, por lo que los Nodos se pusieron sobre la hilera del medio, a aproximadamente 3 metros de cada pared a lo largo, es decir, hay una distancia de 9 metros entre el Nodo 1 y Nodo 2, y a 3 metros de alto. En la Figura 5-8 y 5-9 se observa el diagrama de la ubicación de los Nodos en este invernadero, así se puede comprender de mejor forma el lugar donde se encuentran estos Nodos. Al igual que el invernadero anterior, la toma de datos se comenzó a realizar cada 10 minutos, sin embargo, se tuvieron algunos problemas de recepción de datos en la calibración debido a un error al cargar el código, y se dejó la toma de datos cada 5 minutos para solucionar más rápido el problema, quedando finalmente con este último valor para las pruebas.

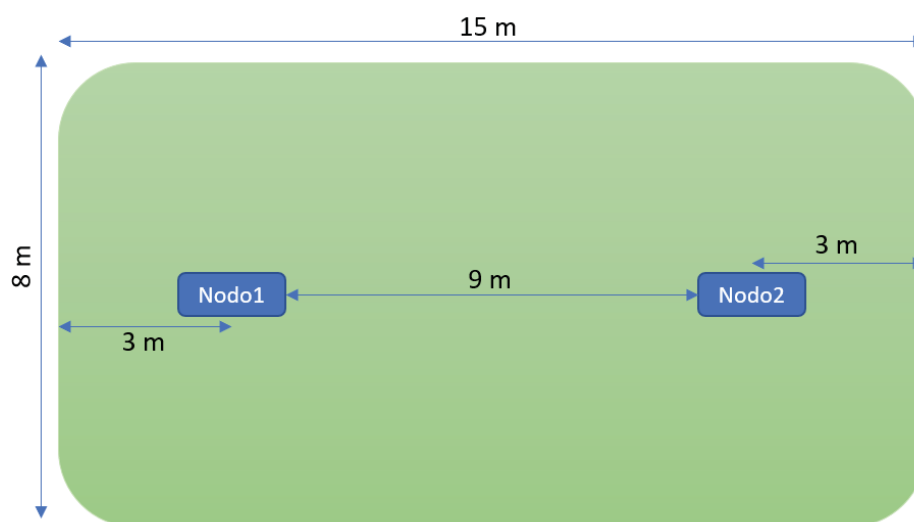


Figura 5-8 Ubicación de Nodos

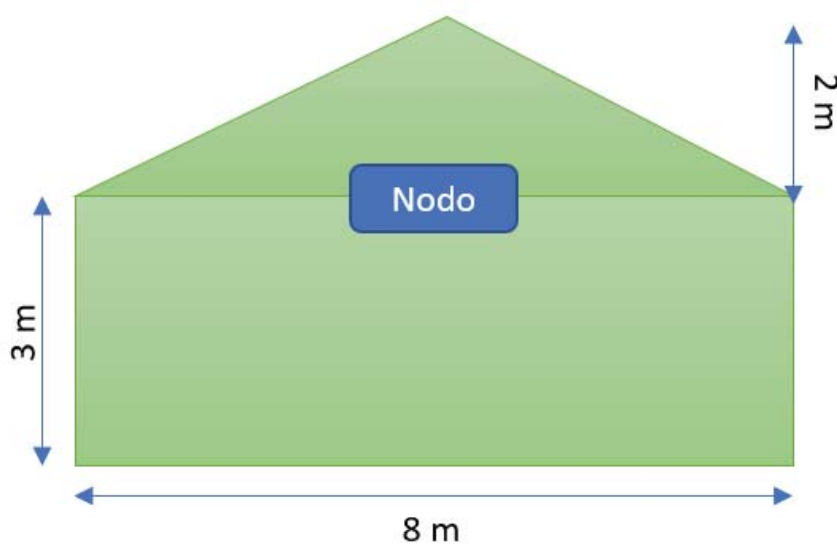


Figura 5-9 Ubicación de Nodos en altura

En este caso la calibración de los sensores de CO₂ quedaron similares al ingresarlos al invernadero y se obtuvo una gráfica similar entre los Nodos, esto depende de la realización de la calibración y que tanto se mueva el Nodo al ingresarlo al invernadero.

A continuación, se muestran los resultados obtenidos, los cuales se muestran de la misma forma que el anterior, primero se ve la temperatura y humedad y luego la luminosidad y CO₂. Se verá que el Nodo 2 realizó más mediciones, esto se debió a un error al cargar el código al Arduino y luego se solucionó.

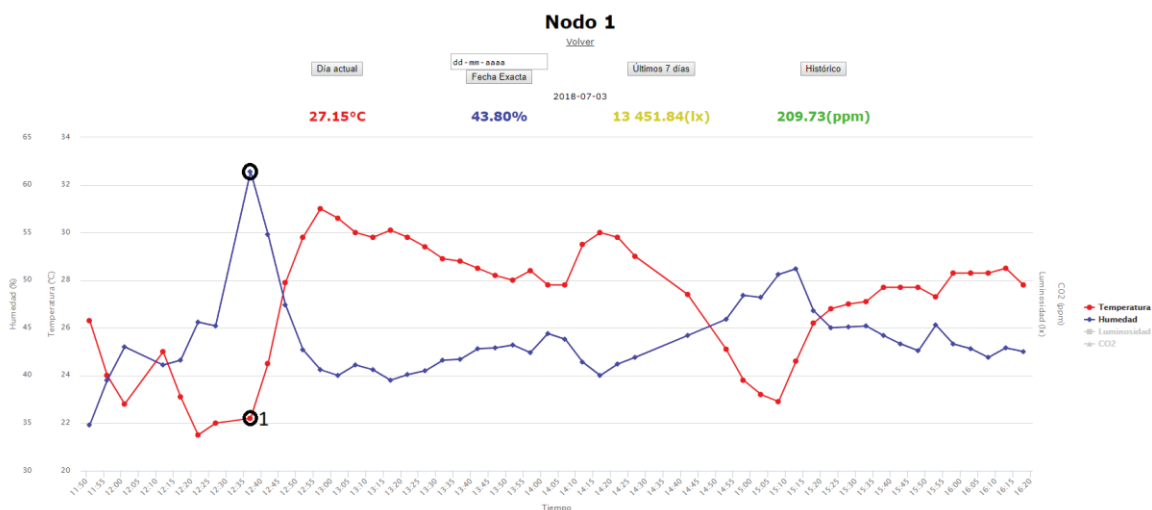


Figura 5-10 Temperatura y humedad Nodo 1

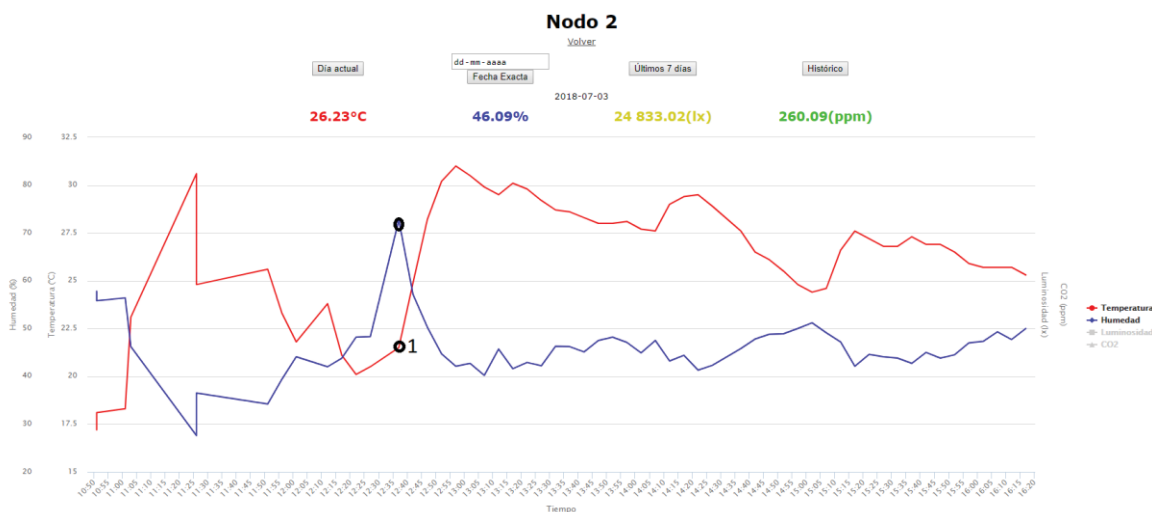


Figura 5-11 Temperatura y humedad Nodo 2

En la Figura 5-10 y 5-11 se puede observar que se tiene un ambiente mucho más homogéneo que el anterior, ya que se tienen gráficas y temperatura y humedad promedio prácticamente iguales en ambos Nodos, lo cual se debe a la calidad del invernadero y a la altura de este, lo cual influye considerablemente en las variables. Se observan gráficas bastante variables, esto es normal de un

día nublado, donde sale y se esconde el sol a ratos. En el punto 1 se ingresan los Nodos al invernadero, donde se observa que la temperatura aumenta en aproximadamente 10 °C lo cual es normal, y la humedad disminuye en aproximadamente un 30% debido a que el volumen del cultivo no es tan grande y se tiene una menor transpiración. Debido a que el día estaba nublado y con una temperatura promedio de 15 °C el invernadero permaneció cerrado todo el día.

Desde las 14:42 hasta el final se comenzaron a mover los Nodos para ver las variaciones que hay al interior. Se pusieron los Nodos en las esquinas, donde la temperatura disminuyó aproximadamente 5 °C y la humedad aumentó aproximadamente 9 % debido a que hay algunas fugas en las paredes y había mucho viento. Al poner los Nodos más hacia el centro, las variables se mantienen relativamente constantes, con menos de 1 °C y 1 %HR de diferencia entre los Nodos.

Como se comentó anteriormente, en este invernadero se tiene un datalogger “HOBO” que mide temperatura y humedad, así, se puede lograr realizar una comparación entre los valores del datalogger y los valores de los Nodos. En la Tabla 5-1 se puede observar la comparación de los datos del datalogger con los datos de los Nodos, considerando el horario desde que se ingresaron los Nodos al invernadero, 12.30, hasta las 14.30. Además, se consideran los valores tomados cada 30 minutos debido a que este es el periodo de toma de datos del datalogger.

Tabla 5-1 Comparación datos Hobo y Nodos

Hora	Temperatura Hobo	Temperatura Nodos	Humedad Hobo	Humedad Nodos
12.30	28.468	22	58.189	48.3
13.00	27.554	31	56.105	42.1
13.30	25.647	29.4	62.151	46.3
14.00	24.823	28.4	63.209	47.1
14.30	23.761	29	60.394	42.3

La diferencia de temperatura entre el datalogger y los Nodos es de aproximadamente 5 °C, esta diferencia se debe a que el datalogger tiene el sensor cubierto para que no le llegue el sol directamente y los Nodos tienen mayor temperatura por el material y color de la carcasa, lo cual genera más calor. La diferencia de humedad relativa es de aproximadamente 15 %, donde el datalogger presenta una mayor humedad debido a que este se encuentra ubicado al medio de dos cultivos distintos: paltos, que es en donde se realizaron las mediciones, y cítricos. Los cítricos tienen un mayor volumen que los paltos, por lo que se tiene más transpiración y genera mayor humedad.

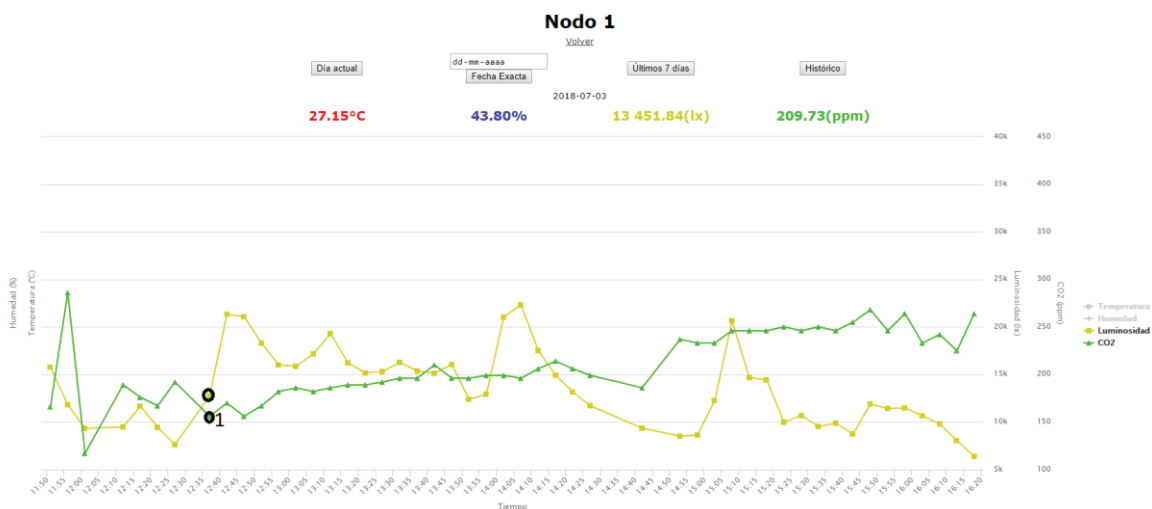


Figura 5-12 Luminosidad y CO2 Nodo 1

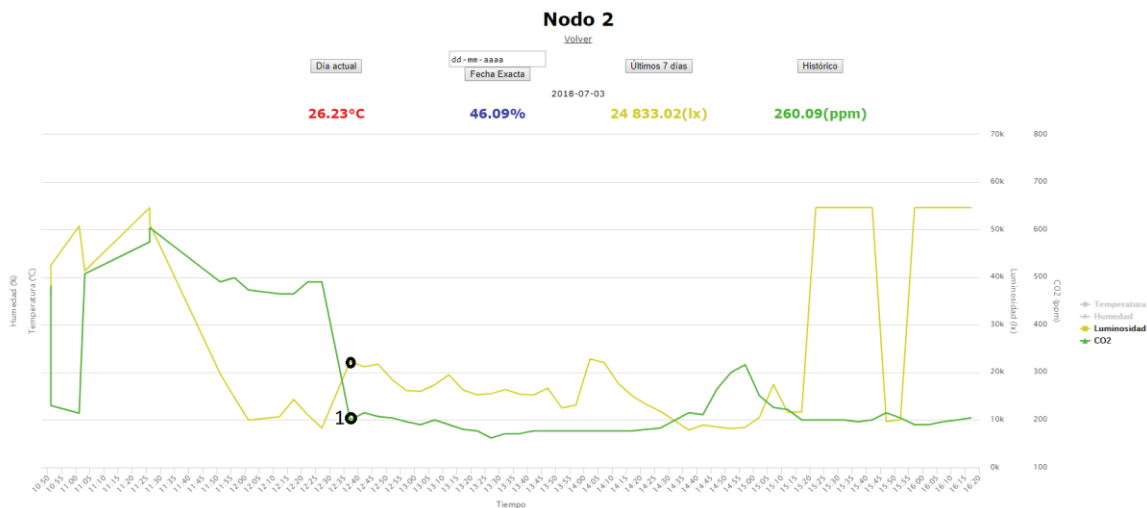


Figura 5-13 Luminosidad y CO2 Nodo 2

Como se puede observar en la Figura 5-12 y 5-13, la luminosidad y CO₂ presentan gráficas similares en ambos Nodos al interior del invernadero. La luminosidad promedio del Nodo 2 no representa realmente lo que pasó en el ambiente debido a que ocurrió un error en la medición y marcó un valor muy grande. El CO₂ promedio de los Nodos tienen valores relativamente similares y la diferencia se debe mayormente a la calibración de los sensores, sin embargo, el Nodo 2 quedó mejor calibrado y se considerará este para sacar conclusiones. En el punto 1, al ingresar a los invernaderos, se tiene una baja de CO₂, al igual como ocurrió en el invernadero de Limache, y donde presenta un alta, se debe a que en ese momento se puso el sensor al lado de la pared donde se tiene una fuga y mayor circulación de aire. La luminosidad varía bastante debido a que salía el sol y luego se escondía, esta vez se realizaron las mediciones sin la tapa, por esto es que se tiene una gráfica muy similar entre ambos Nodos.

5.4 Incorporación alarma

Una vez que se probó y verificó el correcto funcionamiento del sistema de monitoreo, se le incorpora el sistema de alarma, de forma que funcionen ambos sistemas en conjunto. Para incorporar la alarma vía E-mail solo basta realizar la programación en Python que permita en envío. Para probar el sistema de alarma y corroborar que funciona correctamente se realizaron algunas pruebas, modificando los límites de las variables. Para ver que funciona correctamente este Sistema, se toman los datos para ver cuáles son los valores y así poder determinar los límites para el ambiente actual. Luego se fueron variando los límites para que se salieran los valores y se enviara el correo. Finalmente, se obtiene exitosamente el E-mail, donde indica si hay una o más variables fuera de los límites determinados anteriormente.

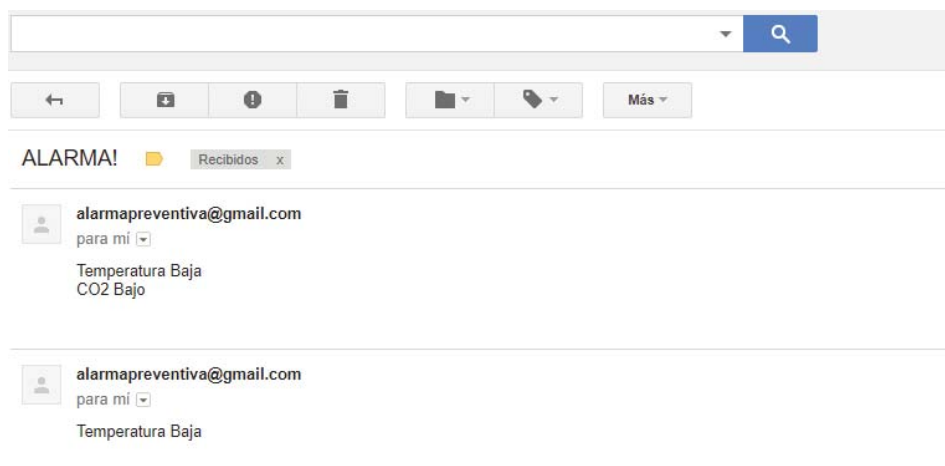


Figura 5-14 Primera prueba de Alarma a correo electrónico

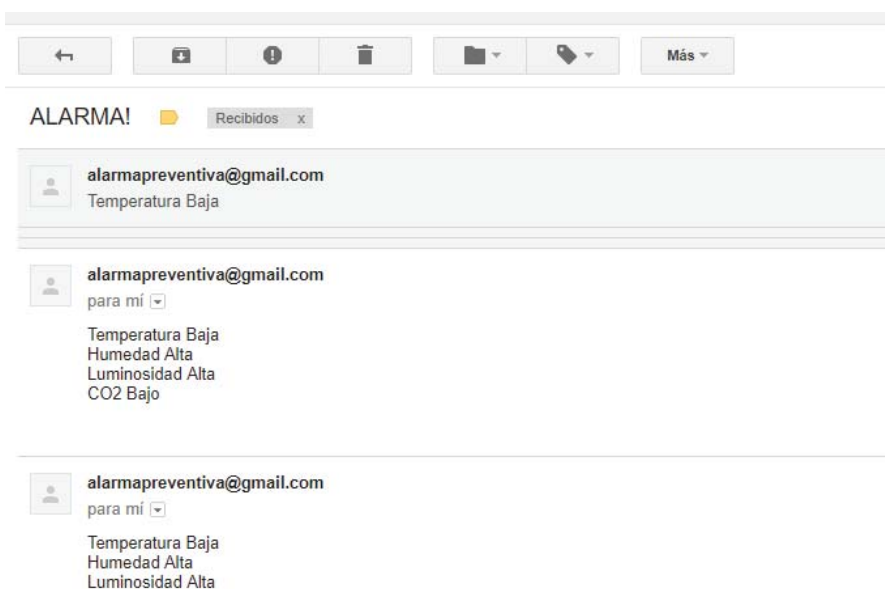


Figura 5-15 Segunda prueba de Alarma a correo electrónico

Se realizaron varias pruebas variando los límites de las variables para ver que funciona correctamente. En la Figura 5-14 se modificó el límite de temperatura y CO₂, de forma de obtener una alarma en ambos casos indicando cuál es la variable que está teniendo problemas. En la Figura 5-15 se modificaron las otras dos variables, obteniendo una alarma con todas las variables con problemas. Además, si ninguna variable se ha salido de sus límites, no se envía ninguna alarma. Con esto, se puede concluir que la alarma por medio de correo electrónico funciona correctamente y se incorpora sin problemas al Sistema de Monitoreo

6 Evaluación Económica

Una parte importante del desarrollo de un proyecto es realizar un estudio económico, de esta forma se tendrá conocimiento de las necesidades económicas que este proyecto requiere, además, quedará claro si el proyecto es rentable o no. Es un estudio sistemático y estructurado para determinar el valor monetario de toda la parte técnica que se ha visto. Para demostrar la rentabilidad del proyecto, se utilizarán indicadores como el VAN y el TIR.

6.1 Componentes

Para realizar el análisis económico, se debe estimar el costo que tiene el Nodo “Dispositivo Final” y el Nodo “Coordinador”, para esto es necesario ver los componentes que posee cada Nodo. En la Tabla 6-1, se muestra la tabla de componentes utilizados para la confección de ambos Nodos.

Tabla 6-1 Tabla de componentes de los Nodos

Nodo “Dispositivo Final”	Nodo “Coordinador”
Arduino Uno	Raspberry Pi 2B
Shield XBee Arduino	XBee Explorer USB
XBee	XBee
DHT-22	Carcasa 3D
BH-1750	Pantalla
MG-811	Mouse
Protoboardb 400 puntos	Teclado
Carcasa 3D	

6.2 Mercado de proveedores

Hoy en día existen diversas empresas que ofrecen componentes electrónicos y que pueden ser nuestros proveedores. Se verán algunos que están presentes en Chile para ver dónde es más conveniente comprar.

Tabla 6-2 Proveedores

Componente	Olimex	Maxelectrónica	Tectronix	Victronics	Pc Factory	Cintegral
Arduino Uno	\$13.860	\$9.990	\$12.000	\$10.450	-	-
Raspberry Pi 2B	-	-	\$26.000	-	-	-
XBee	\$22.800	-	-	\$24.900	-	-
XBee Explorer USB	\$19.990	\$5.990	\$5.800	\$11.600	-	-
Shield XBee Arduino	-	\$6.990	\$11.500	\$8.500	-	-
DHT-22		\$7.490	\$9.000	\$3.990	-	-
BH-1750		-	-	\$4.650	-	-
MG-811	\$49.900	-	-	-	-	-
Protoboard	\$2.290	\$2.490	\$2.300	-	-	-
Pantalla	-	-	-	-	\$49.990	\$65.990
Mouse	-	-	-	-	\$2.990	\$2.490
Teclado	-	-	-	-	\$4.990	\$3.990

De la Tabla 6-2 se obtienen los precios de los componentes necesarios para la realización de ambos Nodos. El valor de la carcasa 3D se considera con el valor del filamento y la cantidad de horas que trabajó la impresora, por lo que se tiene un valor por Nodo de aproximadamente \$15.000.

6.3 Análisis de costos

Una vez que se tienen los proveedores y los precios de cada componente, se puede determinar el precio total de cada Nodo. Además, se debe determinar la cantidad de Horas-Hombre trabajadas para la realización de este proyecto, donde se consideran 8 meses de trabajo, donde se trabajaron 15 horas semanales. En la Tabla 6-3 se muestra el valor a considerar de cada componente en cada Nodo y su valor final, mientras que en la Tabla 6-4 muestra la cantidad de horas que se dedicaron al proyecto y su valor monetario.

Tabla 6-3 Costo por Nodo

Nodo "Dispositivo Final"		Nodo "Coordinador"	
Arduino Uno	\$9.990	Raspberry Pi 2B	\$26.000
Shield XBee Arduino	\$6.990	XBee Explorer USB	\$5.800
XBee	\$22.800	XBee	\$22.800
DHT-22	\$3.990	Carcasa 3D	\$15.000
BH-1750	\$4.650	Pantalla	\$49.990
MG-811	\$49.900	Mouse	\$2.490
Protoboard 400 puntos	\$2.290	Teclado	\$3.990
Carcasa 3D	\$15.000		
Total:	\$115.610	Total:	\$126.070

Tabla 6-4 Costo Horas-Hombre

Actividad	Horas-Hombre	Valor Hora-Hombre	Total
Investigación y desarrollo	480	\$8000	\$3.840.000

6.4 Estudio de Rentabilidad

Una vez determinados todos los costos se puede determinar la inversión inicial necesaria para este proyecto, para esto se debe tener en consideración que para un invernadero se deben instalar 4 Nodos “Dispositivo Final” y un Nodo “Coordinador”. Además, generalmente los clientes no poseen sólo un invernadero, por lo que se considerará un promedio de 5 invernaderos por cliente, es decir, 20 Nodos “Dispositivo Final” y sólo un Nodo “Coordinador”. Así, el costo del proyecto por cliente es de \$2.438.270, por lo que se considerará un valor de venta de \$3.100.000, de modo de tener una ganancia de aproximadamente 30% en componentes. Para realizar el estudio de rentabilidad se estima la instalación del producto a 5 clientes en el primer año, por lo que se tiene una inversión inicial de componentes de \$12.191.350. (Tabla 6-5)

Tabla 6-5 Inversión inicial

Concepto	Total
Componentes	\$12.191.350
Investigación y desarrollo	\$3.840.000
Total	\$16.031.350

Para la implementación del proyecto es necesario contratar a un técnico que realice la instalación de los dispositivos en el invernadero, considerando que no se requieren grandes implementaciones, se considerará un valor por hora de \$4.000 y un total de 5 horas por cliente. A continuación, en la Tabla 6-7 se realizará el flujo de caja, donde se considerarán los datos mostrados en la Tabla 6-6.

Tabla 6-6 Valores necesarios para Flujo

Concepto	Precio
Costo del sistema	\$2.438.270
Valor de venta del sistema	\$3.100.000
Impuesto	20%
TRMA (tasa de retorno)	18%
Técnico	\$20.000

Tabla 6-7 Flujo de Caja

	0	1	2	3	4	5
Cant. clientes		5	10	10	20	20
Inversión	-16031350					
Ingreso venta		15500000	31000000	31000000	62000000	62000000
Costo prod.		-12191350	-24382700	-24382700	-48765400	-48765400
Costo técnico		-100000	-200000	-200000	-400000	-400000
Ut. Ant. Imp.		3208650	6417300	6417300	12834600	12834600
Impuesto		-641730	-1283460	-1283460	-2566920	-2566920
Utilidad		2566920	5133840	5133840	10267680	10267680
Flujo Neto	-16031350	2566920	5133840	5133840	10267680	10267680

Para que el proyecto sea rentable se debe obtener el VAN > 0 y el TIR > TRMA. Del flujo de caja de la Tabla 6-7 se obtiene que el VAN es de \$7.969.966 y el TIR del 24%, lo cual indica que este proyecto es económicamente rentable y realizable.

6.5 Comparación con mercado actual

Al comienzo de este proyecto, en la investigación, se realizaron algunas cotizaciones de sistemas de monitoreo de temperatura y humedad para invernaderos que se encuentran en el mercado actual en Chile. La empresa Epsilon es la que más se acerca a lo que este proyecto ofrece teniendo dos opciones, de forma remota o directamente a la nube, siendo el primero el que se adapta mejor a este proyecto.

Asumiendo que los costos de instalación son muy similares a los considerados en este proyecto, se puede realizar una comparación del costo que tiene la implementación del sistema que ofrece la empresa Epsilon con la implementación de este proyecto, considerando los valores del equipo.

Se considera el supuesto de la instalación de 4 Nodos “Dispositivo Final” por invernadero y en promedio 5 invernaderos por cliente, se tendrá que instalar 20 Nodos “Dispositivo Final” y un Nodo “Coordinador”. En el caso de Epsilon, para obtener la misma cantidad de Nodos que el proyecto se debe considerar 20 equipos de medición (temperatura, humedad y CO₂), un equipo inalámbrico que hará el puente de comunicación y 2 Datalogger que permitirán la visualización de los datos, se requieren 2 debido a que un Datalogger soporta hasta 16 equipos de medición.

En la Tabla 6-8 se muestran los valores unitarios de los Nodos que utiliza este proyecto y de los equipos que utiliza el sistema de la empresa Epsilon.

Tabla 6-8 Valores unitarios dispositivos y equipos

Proyecto	Valor unitario	Epssilon	Valor Unitario
Nodo "Dispositivo Final"	\$115.610	Equipo de medición	\$142.350
Nodo "Coordinador"	\$126.070	Equipo Inalámbrico	\$129.350
		Datalogger	\$291.850

Considerando 20 Nodos "Dispositivo Final" y 1 Nodo "Coordinador", se tendrá un valor total por cliente de \$2.438.270, sin embargo, para comparar con el mercado, se tendrá que considerar el valor de venta, es decir, \$3.200.000. Mientras que para Epssilon, al considerar 20 equipos de medición, 1 equipo inalámbrico y 2 Datalogger se tendrá un valor total por cliente de \$ 3.560.050.

Tabla 6-9 Valor total por cliente

	Valor por cliente
Proyecto	\$3.100.000
Epssilon	\$3.560.050

Por lo tanto, de la Tabla 6-9 se puede concluir que este proyecto está dentro de los valores del mercado, incluso es más económico, por lo que podría insertarse sin problemas en el mercado y comenzar a competir con las grandes empresas. Además, se tiene en cuenta que con el valor que se considera se tiene una ganancia de aproximadamente 30%, por lo que si llegara a ser necesario disminuir este valor para entrar fuertemente al mercado podría ser posible.

Discusión y conclusiones

Los sistemas de monitoreo automatizados están tomando cada vez más fuerzas en distintas áreas, ya que se hace necesario obtener información con más exactitud, cosa que el hombre no logra. Una de las áreas donde se requiere el monitoreo de distintas variables es en agricultura, específicamente, invernaderos, ya que se requieren condiciones especiales para poder producir adecuadamente. Sin embargo, los sistemas de monitoreo automatizados son muy costosos. Por lo que se hace necesaria la creación de un sistema de monitoreo automatizado en invernaderos a un menor costo para que las pequeñas y medianas empresas puedan tener acceso a estos, así, podrían tener la opción de entrar al mercado a competir debido a que el cultivo producido tendrían las características necesarias para hacerlo.

Si bien, los sistemas de monitoreo automáticos son muy útiles para obtener información precisa, no son tan útiles al momento de la toma de decisiones, ya que no indica si se debe actuar o no en un momento determinado, solo entrega la información requerida. Por lo que agregarle un sistema que permita realizar una acción y tomar una decisión ayuda a tener un sistema más completo y eficiente. Un sistema capaz de hacer eso es el sistema de alarmas, el cual se basa en la información obtenida en el monitoreo, pero permite tomar una decisión a través de la recepción de una alarma, la cual indicará sobre alguna situación anormal.

Actualmente existen diversos sistemas de monitoreo y alarma, donde va variando la forma de recibir la información y poder visualizarla y el medio de envío de la alarma. Hay sistemas más antiguos donde el monitoreo y la alarma se obtienen directamente en el lugar, lo cual limita mucho el sistema, mientras que hay otros sistemas más actuales donde el monitoreo y alarma se obtienen en cualquier parte del mundo, dando gran flexibilidad al sistema.

De todas las opciones vistas, el que más se adapta a este proyecto es el monitoreo inalámbrico vía IP y alarma vía E-mail. El monitoreo inalámbrico es de gran importancia debido al ambiente donde se encuentra inserto el invernadero, donde generalmente están en lugares alejados y se requiere un sistema a distancia y así no tener la necesidad de encontrarse directamente en el invernadero para visualizar el monitoreo y, una vez que se tiene el sistema a distancia, la información podrá ser visualizada en cualquier dispositivo conectado a la misma red que el sistema gracias al monitoreo vía IP, además, la alarma se podrá recibir en cualquier parte.

Para lograr la implementación del sistema de monitoreo y alarma fue necesario estudiar los distintos componentes posibles a utilizar, llegando a un diseño final del sistema que incluye módulos XBee, Raspberry Pi, Arduino y sensores de las distintas variables importantes a monitorear, estas variables se determinaron realizando un estudio de distintos cultivos, concluyendo que existen cuatro variables de gran importancia: Temperatura, Humedad Relativa, Dióxido de Carbono y Luminosidad. El diseño del sistema incluye dos tipos de dispositivos denominados Nodo “Dispositivo Final” y Nodo “Coordinador”, donde cada uno se compone de distintos dispositivos y tienen funciones específicas, el primero se encarga de la obtención y envío de datos, mientras que el segundo se encarga de recibir, guardar y organizar la información para ser visualizada y, además, trabajar con esta información para la obtención de la alarma.

Para lograr la comunicación entre Nodo “Dispositivo Final” y Nodo “Coordinador” se utilizaron los módulos XBee, de forma de obtener una conexión inalámbrica. Para la utilización de estos módulos fue necesario realizar la configuración necesaria para lograr la comunicación. El Nodo “Dispositivo Final” sólo se encargará de obtener los datos y enviarlos, por lo que los módulos se configuraron en modo transparente, mientras que el Nodo “Coordinador” se encarga de procesar toda la información, por lo tanto, será necesario recibir la información de forma ordenada, de esta forma, el módulo se debió configurar en modo API.

Un dispositivo de gran importancia en este sistema de monitoreo y alarma es la placa Raspberry Pi, ya que será el cerebro del sistema, permitiendo captar toda la información y trabajar con ella. Este dispositivo posee una gran versatilidad y se adapta a una inmensidad de aplicaciones, por lo que se puede utilizar para diversas funciones, en este caso se utilizó para recibir la información de forma inalámbrica a través del módulo XBee y como servidor web para permitir la visualización de los datos de forma fácil y cómoda para el usuario, además, permitirá trabajar con los datos y lograr la realización de la alarma. Para lograr estas funcionalidades se debió realizar la programación necesaria en dos lenguajes de programación, Python y PHP.

Una vez que se logró el envío y recepción de los datos en los Nodos, se pudo comenzar a trabajar con los datos, donde fue necesaria la realización de una Base de Datos, la cual permitirá almacenar los datos de los sensores de forma ordenada en una tabla, indicando la fecha y hora en que se recibió cada dato. Esto permitirá realizar la Interfaz gráfica que verán los clientes, donde muestra los datos de la Base de Datos en gráficos para poder realizar cualquier análisis que necesite el cliente.

Finalmente, se realizó la Interfaz Gráfica que permitirá la visualización mediante una página web, en esta página se tiene un menú principal con opción a visualizar los datos del Nodo 1 o Nodo 2, luego, al ingresar a algún nodo, se visualizan los gráficos de los distintos variables, teniendo la opción de ver los datos de distintos periodos de tiempo, lo cual permite que esta interfaz sea más útil al momento de necesitar visualizar algún día específico o la última semana. Además de los gráficos, se muestra el último dato recibido o el promedio de los datos. Esta interfaz permite que el monitoreo sea mucho más simple para el cliente y permitiéndole realizar distintos análisis con los datos y gráficos. Además, se tiene la opción de exportar los datos, lo cual es muy útil en muchos casos.

Una vez realizado el prototipo final del sistema de monitoreo y alarma, se realizaron las pruebas en terreno, es decir, en invernaderos, con el fin de validar el prototipo. Las pruebas se realizaron en dos invernaderos, donde se obtuvieron resultados similares, llegando a la conclusión que este sistema funciona bastante bien en el ambiente para el que fue realizado, obteniendo resultados dentro de los rangos esperados. La única variable de la que no se obtuvo un valor representativo fue el de CO₂, donde la calibración no se pudo realizar eficientemente y no se pudo determinar si el valor estaba realmente correcto, sin embargo, se puede observar el comportamiento de esta variable, donde va a aumentar o disminuir dependiendo el estado en que se encuentre el invernadero. Por lo tanto, en este sistema se podrá obtener una representación gráfica de lo que está sucediendo con el CO₂, pero no indicar el valor exacto de este.

En las pruebas en terreno se pudo determinar la cantidad necesaria de Nodos “Dispositivo Final” al interior de los invernaderos, teniendo en cuenta que son invernaderos de mediciones estándar, llegando a la conclusión que se requerirán 3 o 4 Nodos dependiendo la calidad del invernadero y de la homogeneidad del ambiente interno. Sin embargo, quedó claro que se deben realizar las calibraciones y evaluaciones necesarias en cada invernadero donde se quiera implementar este sistema debido a que el ambiente puede ser muy diferente en uno y otro invernadero.

Además, se pudo observar que es muy importante que la alimentación del Nodo “Dispositivo Final” cumpla con los requerimientos del nodo, debido a que si no cumple con esto se tendrán constantes pérdidas de datos debido a que el módulo XBee no tiene la potencia necesaria para realizar el envío de información.

Por otra parte, se probó el sistema de monitoreo con la incorporación de las alarmas preventivas, donde se pudo comprobar su correcto funcionamiento donde, al salirse los valores de los límites establecidos, se activa la alarma y se envía un mensaje vía E-mail indicando la variable que está fuera del rango y si está bajo o sobre éste.

De esta forma se pudo determinar que el sistema es funcional en su ambiente y, además, muy atractivo para los demás debido a que se pueden visualizar los datos a tiempo real, generándose automáticamente los gráficos, todo esto permite un gran ahorro de tiempo permitiendo evitar futuros problemas. Los encargados de los invernaderos donde se realizaron las pruebas quedaron muy interesados con este sistema de monitoreo y alarma debido a que pudieron observar que es cómodo de utilizar gracias a que es a distancia y por mostrar los datos a tiempo real. Uno de los invernaderos tenía incorporado un sistema de medición, el cual es un Datalogger de donde se extraen los datos una vez a la semana, donde se extrae una tabla de datos y luego el encargado de este invernadero debe realizar los distintos gráficos, lo cual le lleva mucho tiempo. Este al ver el proyecto quedó encantado porque le ahorraría mucho tiempo y, además, al tener la alarma por medio de E-mail, si no se encuentra en el lugar, podrá enviar a alguien para que realice alguna acción al recibir la alarma.

Se realizó un análisis económico para determinar el costo que tiene el proyecto y su rentabilidad en un periodo de 5 años, de esto se pudo concluir que el proyecto es rentable con los supuestos realizados. Además, se comparó el sistema con los valores del mercado y se pudo observar que se encuentra bajo el valor que otras empresas poseen para un sistema similar a este.

Si bien, este sistema de monitoreo y alarma cumple con los objetivos y tiene gran funcionalidad, se puede realizar diversos cambios que permitirán obtener un sistema más completo y representativo de los invernaderos. El cambio principal será en el sensor de Dióxido de Carbono, el cual funciona, pero no como se quisiera, por lo que habría que ver la opción de mejorar la calibración o buscar otro sensor que cumpla con los requerimientos. Además, al realizar las pruebas en terreno, los expertos del tema indicaron que en vez de medir la luminosidad es mejor medir la radiación debido a que la radiación es de gran importancia para la fotosíntesis, no tanto así la luminosidad. Por otra parte, se obtuvieron algunas diferencias de temperatura al comparar con el Datalogger que se encontraba en uno de los invernaderos, lo cual se debió principalmente a que la carcasa es de color negro, lo cual genera más calor y a que el sensor de temperatura no tiene una cubierta y le llega directamente el sol, por lo que habría que considerar proteger los sensores con algún material adecuado.

Por otra parte, este proyecto se puede completar mucho más integrando, por ejemplo, el control de las variables. Ya se cuenta con el monitoreo de las variables y las alarmas preventivas, pero sería interesante continuarlo para lograr un producto final mucho más completo y útil, dejando el sistema completamente automático, desde la medición de las variables hasta controlar cada variable. Para la implementación del control se requerirá realizar una gran inversión en un invernadero para poder adaptarlo e implementar los actuadores, es por esto que muchos sistemas no poseen esta opción, ya que es muy costoso y requiere de un gran trabajo tanto de adaptación del invernadero como la realización de este. Además, para la implementación del control es necesario evaluar tanto las condiciones internas y externas del invernadero, ya que dependerá en gran parte del flujo de aire y cómo circulará este desde fuera hacia adentro, por lo que se debe tener en cuenta que se deben agregar distintos sensores para realizar las mediciones fuera del invernadero.

Bibliografía

- [1] Infoagro, «Control climático en invernaderos,» [En línea]. Available: http://www.infoagro.com/industria_auxiliar/control_climatico4.htm.
- [2] A. Torres, «Manual de cultivo del tomate bajo invernadero,» INIA, 2017.
- [3] «Tomate al CO2,» 24 enero 2010. [En línea]. Available: <http://antonuriarte.blogspot.com/2010/01/tomate-al-co2.html>.
- [4] U. Vegas y M. Narrea, «Manejo integrado del cultivo de limón,» Agrobanco, 2011.
- [5] C. Royo, «Respuesta de los frutos cítricos a las bajas temperaturas: estudio mediante micromatrices,» 2010.
- [6] «El cultivo del palto,» INIA, 2010.
- [7] DANE, «El cultivo del pimentón bajo invernadero,» 2015.
- [8] J. R. Mármol, «Cultivo del pimiento dulce en invernadero,» 2010.
- [9] Solsis, «Controlador Invernador,» [En línea]. Available: <http://www.solsis.cl/invernad.htm>.
- [10] JMI, «Control GSM 5010 5011,» [En línea]. Available: <https://www.jmi.com.mx/control-celular-5010-5011.html>.
- [11] EtherPower, «SUN-32».
- [12] R. H. Weber, «Internet of Things- New Security and Privacy Challenges,» 2010.
- [13] J. C. Pulgarín, «Funcionamiento de un sistema de alarmas».

-
- [14] B. Engineering, «Sure Cross DX80 FlexPower Node».
- [15] J. Contreras, «Integración de placa Raspberry Pi con Shield XBee,» Valparaíso, 2017.
- [16] «¿QUÉ ES XBEE?,» [En línea]. Available: <http://xbee.cl/que-es-xbee/>.
- [17] Digi, «XBee/XBee-PRO S2C 802.15.4, User Guide,» 2016.
- [18] Digi, «XBee ZigBee mesh kit, User Guide,» 2016.
- [19] «Zig Bee/ XBee,» 16 Noviembre 2016. [En línea]. Available: <https://aprendiendoarduino.wordpress.com/2016/11/16/zigbeexbee/>.
- [20] L. Llamas, «¿Que es Raspberry Pi?,» 15 Octubre 2017. [En línea]. Available: <https://www.luisllamas.es/que-es-raspberry-pi/>.
- [21] «Raspberry Pi 2 Model B,» [En línea]. Available: <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>.
- [22] «GPIO,» [En línea]. Available: <https://www.raspberrypi.org/documentation/usage/gpio/README.md>.
- [23] «¿Qué es Arduino?,» [En línea]. Available: <http://arduino.cl/que-es-arduino/>.
- [24] A. N. Vegas, «ARDUINO UNO».
- [25] «Qué es un sensor,» [En línea]. Available: http://www.profesormolina.com.ar/tecnologia/sens_transduct/que_es.htm.
- [26] T. Liu, «DHT22,» [En línea]. Available: <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>.
- [27] M. Electronics, «Digital 16bit Serial Output Type Ambient Light Sensor IC,» [En línea]. Available: <https://www.mouser.com/ds/2/348/bh1750fvi-e-186247.pdf>.
- [28] «MG811 CO2 Sensor,» [En línea]. Available: <https://sandboxelectronics.com/files/SEN-000007/MG811.pdf>.
- [29] «Arduino 1.8.5,» [En línea]. Available: <https://www.arduino.cc/en/main/software#>.
- [30] «¿Qué es Python?,» 10 Enero 2008. [En línea]. Available: <http://mundogeek.net/archivos/2008/01/10/%C2%BFque-es-python/>.

-
- [31] «Qué es y para qué sirve MySQL,» [En línea]. Available: <http://culturacion.com/que-es-y-para-que-sirve-mysql/>.
- [32] Carlos, «¿Qué es phpMyAdmin?,» 4 enero 2012. [En línea]. Available: <https://www.hostname.cl/blog/que-es-phpmyadmin>.
- [33] E. F. Cases, «¿Qué es PHP?¿Para qué sirve?,» 21 octubre 2014. [En línea]. Available: <http://www.ibrugor.com/blog/que-es-php-para-que-sirve/>.
- [34] D. Pineda, «Servidor Web,» 2014.
- [35] Innervycs, «XBee Modo API y Comunicación».
- [36] E. Bonet, «Servicios de acceso remoto II: SSH,» [En línea]. Available: <http://informatica.uv.es/it3guia/AGR/apuntes/teoria/documentos/SSH.pdf>.
- [37] «VNC User Guide,» diciembre 2015. [En línea]. Available: https://www.realvnc.com/en/connect/_downloads/VNC_User_Guide.pdf.
- [38] M. Vergara, «Teoría de Baterías,» [En línea]. Available: <http://www2.elo.utfsm.cl/~elo383/apuntes/InformeBaterias.pdf>.
- [39] Digi, «Digi XBee Python library,» [En línea]. Available: <https://github.com/digidotcom/python-xbee>.
- [40] M. Angel, «¿Qué es LAMP?,» [En línea]. Available: <https://desarrolloweb.com/faq/472.php>.
- [41] «Install Apache, PHP7.0 and MySQL/MariaDB on a Raspberry Pi (LAMP) with phpMyAdmin,» 2 noviembre 2017. [En línea]. Available: <http://www.pihome.eu/2017/11/02/install-apache-php-7-0-mysqldb-raspberry-pi-lamp/>.
- [42] «Conectarse a MySQL con Python,» 3 julio 2013. [En línea]. Available: <https://www.todavianose.com/conectarse-a-mysql-con-python/>.
- [43] A. G. González, «Gráficas Web en tiempo real con CanvasJS,» 21 abril 2016. [En línea]. Available: <http://panamahitek.com/graficas-web-tiempo-real-canvasjs/>.
- [44] «Enviar email desde Python con adjuntos Gmail,» 14 diciembre 2017. [En línea]. Available: <http://www.mnp.cl/post/enviar-email-python-raspbian-gmail-adjunto>.
- [45] V. H. B. Avelar, «Guía técnica del cultivo del aguacate,» 2003.

A Programación

Para la realización del sistema de monitoreo y alarma es necesario utilizar distintos lenguajes de programación como lo son Arduino, Python y PHP. Cada lenguaje tiene un objetivo específico y en conjunto se obtendrá el sistema completo.

A.1 Arduino

Para la lectura de los sensores y luego el envío de estos datos se requiere realizar la programación necesaria en Arduino.

```
//Iniciar Xbee serial
SoftwareSerial xbee(2, 3);    // RX, TX
//Iniciar Sensor DHT
#define DHTTYPE DHT22
const int DHTPin = 5;
DHT dht(DHTPin, DHTTYPE);
//Inicial sensor BH1750
BH1750 lightMeter;
//Iniciar sensor MG811
#define MG_PIN                (A0)
#define DC_GANANCIA           (8.5)
#define ZERO_POINT_VOLTAGE    (0.164)
#define REACTION_VOLTAGE      (0.030)
float CO2Curve[3] = {2.602,ZERO_POINT_VOLTAGE,(REACTION_VOLTAGE/(2.602-3))};
void setup() {
  Serial.begin(9600);        // Baudios monitor serial
  xbee.begin(9600);         // Baudios Xbee
  dht.begin();
  lightMeter.begin();
}
void loop() {
  //valores DHT
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  //valores BH1750
  uint16_t lux = lightMeter.readLightLevel();
  //valores MG811
  float analog = analogRead(MG_PIN);
  float volts = analog*5.0/1024;
  int val = MGGetPercentage(volts,CO2Curve);

  //Creacion del Mensaje
  Serial.print("[[";
  Serial.print(id_disp);
  Serial.print("-");
  Serial.print(h);
  Serial.print("-");
  Serial.print(t);
}
```

```

Serial.print("-");
Serial.print(lux);
Serial.print("-");
Serial.print(val);
Serial.print("-");
Serial.print(volts);
Serial.print("]");
delay(60000);          // Esperar 10 minutos entre mensaje
}
//Funcion para MG811
int MGGetPercentage(float volts, float *pcurve)
{
    return pow(10, ((volts/DC_GANANCIA)-pcurve[1])/pcurve[2]+pcurve[0]);
}

```

A.2 Python

La programación realizada en Python permite recibir, ordenar y almacenar los datos para luego llevarlos a una Base de Datos. Además, se incorpora la alarma en el mismo código.

```

PUERTO = "/dev/ttyAMA0"
BAUD_RATE = 9600
ser = serial.Serial(PUERTO, BAUD_RATE)

xbee = ZigBee(ser)

# datos de base de datos
DB_HOST = 'localhost'
DB_USUARIO = 'root'
DB_CONTR = '1234'
DB_NOMBRE = 'monitoreo'

# Funcion para guardar datos en MySQL
def run_query(query=''):
    datos = [DB_HOST, DB_USUARIO, DB_CONTR, DB_NOMBRE]

    conn = MySQLdb.connect(*datos)#Conectar a base de datos
    cursor = conn.cursor() # Crear un cursor
    cursor.execute(query) # Ejecutar una consulta

    if query.upper().startswith('SELECT'):
        data = cursor.fetchall()
    else:
        conn.commit() # Hacer efectiva la escritura de datos
        data = None

    cursor.close() # Cerrar el cursor
    conn.close() # Cerrar la conexión

# Funcion de envio de correo
def enviar_correo_electronico():
    smtpserver = smtplib.SMTP("smtp.gmail.com", 587)
    smtpserver.ehlo() # prepara envío
    smtpserver.starttls() # Conexión con el servidor SMTP
    smtpserver.login(msg['From'], password) #Inicio de sesión
    msg.attach(MIMEText(mensaje_1, 'plain')) smtpserver.sendmail(msg['From'],
msg['To'], msg.as_string()) # Envío de correo electrónico
    smtpserver.close()

while True:
    response = xbee.wait_read_frame()
    router = response[0]
    humedad = response[1]
    temperatura = response[2]
    luminosidad = response[3]
    CO2 = response[4]

    # envio de variables
    query = ("INSERT INTO datos"

```

```

        "(router, fecha, hora, temperatura, humedad, luminosidad, co2)"
        "VALUES ('%s', '%s', '%s', '%s', '%s', '%s', '%s')") % (
router, fecha, hora, temperatura, humedad, luminosidad, CO2)
run_query(query)

# variables para envio de mensajes
password = " "
msg['From'] = " "
msg['To'] = " "
msg['Subject'] = " "

# alarmas preventivas
if temperatura > "":
    print("Temperatura alta")
    mensaje_1 = ""Temperatura Alta""
    enviar_correo_electronico()

elif temperatura < "":
    print("Temperatura baja")
    mensaje_1 = ""Temperatura Baja""
    enviar_correo_electronico()

```

A.3 PHP

Una vez obtenida la base de datos mediante Python, se puede trabajar con ella para lograr la obtención de una interfaz gráfica, esta interfaz se logra mediante la programación de un código en PHP.

```

<?
//conexión con base de datos
$bd_servidor = "localhost";
$bd_usuario = "root";
$bd_clave = "1234";
$base_de_datos = "monitoreo";
$conexion = mysqli_connect($bd_servidor, $bd_usuario, $bd_clave, $base_de_datos);
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html lang="es">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Interfaz de Monitoreo</title>
<link href="css/style.css" rel="stylesheet">
<script type="text/javascript" src="js/highcharts.js"></script>
<script type="text/javascript" src="js/jquery-1.9.1.js"></script>
<script type="text/javascript" src="js/jquery.min.js"></script>
</head>
<body>
<?
$seccion = $_GET["seccion"];
if (!$seccion) { //menú principal
    echo "<h1>Menú Principal</h1>";
    echo "<a href='index.php?seccion=datos&router=1'>Router 1</a><br>";
    echo "<a href='index.php?seccion=datos&router=1'>Router 2</a><br>";
}
elseif ($seccion == "datos") { //mostrar Datos
    $router = $_GET["router"];
    echo "<h1>Router $router</h1>";
    echo "<a href='index.php'>Volver</a><br>";
    echo "<div id='filtros'>";
    echo "<div id='filtro'>";
    echo "<form method='get' action='index.php'>";
    echo "<input type='hidden' name='seccion' value='datos'>";
    echo "<input type='hidden' name='router' value='$router'>";
    echo "<input type='hidden' name='filtro' value='dia'>";

```

```

echo "<input type='submit' value='Día actual'>";
echo "</form>";
echo "</div>";
echo "<div id='filtro'>";
echo "<form method='get' action='index.php'>";
echo "<input type='date' name='fecha_req'><br />";
echo "<input type='hidden' name='seccion' value='datos'>";
echo "<input type='hidden' name='router' value='$router'>";
echo "<input type='hidden' name='filtro' value='fecha'>";
echo "<input type='submit' value='Fecha Exacta'>";
echo "</form>";
echo "</div>";
echo "<div id='filtro'>";
echo "<form method='get' action='index.php'>";
echo "<input type='hidden' name='seccion' value='datos'>";
echo "<input type='hidden' name='router' value='$router'>";
echo "<input type='hidden' name='filtro' value='7dias'>";
echo "<input type='submit' value='Últimos 7 días'>";
echo "</form>";
echo "</div>";
echo "<div id='filtro'>";
echo "<form method='get' action='index.php'>";
echo "<input type='hidden' name='seccion' value='datos'>";
echo "<input type='hidden' name='router' value='$router'>";
echo "<input type='hidden' name='filtro' value='historico'>";
echo "<input type='submit' value='Histórico'>";
echo "</form>";
echo "</div>";
echo "</div>";
$filtro = $_GET["filtro"];
//si no hay filtro, se muestra el último día
if (!$filtro) {
    $filtro = "dia"; }
//aplicación de los distintos filtros
if ($filtro == "dia") {
    $fecha_hoy = date("Y-m-d");
    echo $fecha_hoy;
    $accion = mysqli_query($conexion, "select * from datos where router='$router' and
fecha='$fecha_hoy' order by id_datos");
}
elseif ($filtro == "historico") {
    echo "Datos Históricos";
    $accion = mysqli_query($conexion, "select * from datos where router='$router' order
by id_datos");
}
//obtención de los datos
while ($fila = mysqli_fetch_array($accion)) {
    $id_datos = $fila["id_datos"];
    $fecha[] = $fila["fecha"];
    $hora[] = $fila["hora"];
    $temperatura[] = $fila["temperatura"];
    $humedad[] = $fila["humedad"];
    $luminosidad[] = $fila["luminosidad"];
    $co2[] = $fila["co2"];
}
echo "<br />";
if ($filtro == "dia") { //día actual
    //últimos datos obtenidos
    $temp = end($temperatura);
    $hum = end($humedad);
    $lum = end($luminosidad);
    $co = end($co2);
}
else { //otras opciones
    //datos promediados
    $temp = array_sum($temperatura) / count($temperatura); //promedio
    $temp = number_format($temp, 2, '.', ' '); //2 decimales
    $hum = array_sum($humedad) / count($humedad); //promedio
    $hum = number_format($hum, 2, '.', ' '); //2 decimales
    $lum = array_sum($luminosidad) / count($luminosidad); //promedio
}

```

```

    $lum = number_format($lum, 2, '.', ' '); //2 decimales
    $co = array_sum($co2) / count($co2); //promedio
    $co = number_format($co, 2, '.', ' '); //2 decimales
}
//se muestran los datos
echo "<div id='datos'>";
echo "<div id='temp'>".$temp."°C</div>";
echo "<div id='hum'>$hum%</div>";
echo "<div id='lum'>$lum(lx)</div>";
echo "<div id='co'>$co(ppm)</div>";
echo "</div>";
//gráfico
echo "<div id='container'></div>";
?>
<script type='text/javascript'>
    Highcharts.setOptions({
        lang: {
            shortMonths: ["Ene", "Feb", "Mar", "Abr", "May", "Jun", "Jul", "Ago", "Sep",
"Oct", "Nov", "Dic"],
            weekdays: ["Domingo", "Lunes", "Martes", "Miércoles", "Jueves", "Viernes",
"Sábado"]
        }
    });
    Highcharts.chart('container', {
        title: {
            text: ''
        },
        yAxis: [{//--- yAxis 1
            title: {
                text: 'Temperatura (°C)'
            }
        }, {//--- yAxis 2
            title: {
                text: 'Humedad (%)'
            }
        }, {//--- yAxis 3
            title: {
                text: 'Luminosidad (lx)'
            },
            opposite: true
        }, {//--- yAxis 4
            title: {
                text: 'CO2 (ppm)'
            },
            opposite: true
        }
    ],
    xAxis: {
        type: 'datetime',
        title: {
            text: 'Tiempo'
        },
        tickInterval: 1
    },
    legend: {
        layout: 'vertical',
        align: 'right',
        verticalAlign: 'middle'
    },
    plotOptions: {
        series: {
            label: {
                connectorAllowed: false
            }
        }
    },
    series: [{
        yAxis: 0,
        name: 'Temperatura',
        color: '#EB1111',
        data: [
</pre>
</div>
<div data-bbox="858 920 889 937" data-label="Page-Footer">87</div>
```



```

<?
  for ($i = 0; $i < mysqli_num_rows($accion); $i++) {
    echo "[Date.UTC(" . date("Y", strtotime($fecha[$i])) . "," . (date("m",
    strtotime($fecha[$i])) - 1) . "," . date("d", strtotime($fecha[$i])) . "," . date("H",
    strtotime($hora[$i])) . "," . date("i", strtotime($hora[$i])) . ")"," . $temperatura[$i] .
    "],";
  }
?>],

  tooltip: {
    valueSuffix: '°C'
  }
}, {
  yAxis: 1,
  visible: false,
  color: '#2518DC',
  name: 'Humedad',
  data: [

<?
  for ($i = 0; $i < mysqli_num_rows($accion); $i++) {
    echo "[Date.UTC(" . date("Y", strtotime($fecha[$i])) . "," . (date("m",
    strtotime($fecha[$i])) - 1) . "," . date("d", strtotime($fecha[$i])) . "," . date("H",
    strtotime($hora[$i])) . "," . date("i", strtotime($hora[$i])) . ")"," . $humedad[$i] . "],";
  }
?>],

  tooltip: {
    valueSuffix: '%'
  }
}, {
  yAxis: 2,
  visible: false,
  color: '#D4CE16',
  name: 'Luminosidad',
  data: [

<?
  for ($i = 0; $i < mysqli_num_rows($accion); $i++) {
    echo "[Date.UTC(" . date("Y", strtotime($fecha[$i])) . "," . (date("m",
    strtotime($fecha[$i])) - 1) . "," . date("d", strtotime($fecha[$i])) . "," . date("H",
    strtotime($hora[$i])) . "," . date("i", strtotime($hora[$i])) . ")"," . $luminosidad[$i] .
    "],";
  }
?>],

  tooltip: {
    valueSuffix: '(lx)'
  }
}, {
  yAxis: 3,
  visible: false,
  color: '#1ECB12',
  name: 'CO2',
  data: [

<?
  for ($i = 0; $i < mysqli_num_rows($accion); $i++) {
    echo "[Date.UTC(" . date("Y", strtotime($fecha[$i])) . "," . (date("m",
    strtotime($fecha[$i])) - 1) . "," . date("d", strtotime($fecha[$i])) . "," . date("H",
    strtotime($hora[$i])) . "," . date("i", strtotime($hora[$i])) . ")"," . $co2[$i] . "],";
  }
?>],

  tooltip: {
    valueSuffix: '(ppm)'
  }
}
}];
//]]>
</script>
<?
}
?>
</body>
</html>

```