

Sistema de control automatizado para pequeños cultivos enfocado en monitoreo y regulación.

Alvaro Luis Rios Garzón y Cristian Guillermo Serrano Acevedo

Trabajo de grado para optar al título de Ingeniero de Sistemas

Director

PhD. Lola Xiomara Bautista Rozo

Doctor en Ciencias y Tecnologías de la comunicación y la Información

Codirector

Urbano Eliécer Gómez Prada

Magíster en Ingeniería Área Informática y Ciencias de la Computación.

Universidad Industrial de Santander

Facultad de Ingeniería Físico-Mecánicas

Escuela de Ingeniería de Sistemas e Informática

Ingeniería de Sistemas

Bucaramanga

2023

Dedicatoria

Este logro quisiéramos dedicárselo a todas esas personas especiales en nuestras vidas que nos han ayudado a crecer, han sido un gran apoyo y nos motivan a seguir adelante, especialmente a María Teresa Acevedo Afanador, Ana María Garzón Plata, Guillermo Serrano Díaz, Luz Marina Plata Plata, Teodoro David Rios Garzón, Cecilia Mercedes Plata Plata Q.E.P.D. y Daniela Cruz Cuevas. También quisiéramos dedicar este logro a nuestros profesores, compañeros y amigos quienes nos acompañaron en este proceso.

Agradecimientos

Quisiéramos agradecer profundamente a nuestros padres, hermanos y abuelos quienes con su apoyo nos ayudaron a superar todos los desafíos que se presentaron a lo largo del desarrollo del proyecto, también quisiéramos dar las gracias a nuestra directora Lola Xiomara Bautista Rozo y a nuestro codirector Urbano Eliécer Gómez Prada por guiarnos y brindarnos su apoyo durante todo el proyecto. A nuestro amigo Pedro Jiménez Buenahora quien nos ayudó en más de una ocasión cuando las cosas se ponían complicadas y también damos gracias a todos nuestros amigos y personas cercanas por ser parte importante en los pequeños aportes al proyecto. No nos caben las palabras para expresar la gratitud que sentimos.

Tabla de Contenido

	Pág.
Introducción	14
1. Descripción del proyecto	16
1.1 Planteamiento y justificación del problema	16
1.2 Metodología	17
2. Objetivos	21
2.1 Objetivo General	21
2.2 Objetivos Específicos.....	21
3. Marco de Referencia	22
3.1 Marco Teórico.....	22
3.2 Marco Conceptual.....	24
3.2.1 Sensor.....	24
3.2.2 Microcontrolador	27
3.2.3 Agricultura	30
3.2.3.1 Cultivos alimentarios	30
3.2.3.2 Cultivos para forraje	30
3.2.3.3 Cultivos textiles	30
3.2.3.4 Cultivos oleaginosos	31
3.2.3.5 Cultivos ornamentales.....	31
3.2.3.6 Cultivos industriales.....	31
3.3 Antecedentes	33
4. Resultados	37

4.1 Requerimientos funcionales y no funcionales	37
4.1.1 Requerimientos funcionales.....	37
4.1.2 Requerimientos no funcionales.....	38
4.2 Diseño de prototipo.....	38
4.2.1 Diagramas de casos de uso	39
4.2.2 Diagramas de actividades	42
4.2.3 Modelo de la base de datos	44
4.2.4 Arquitectura del software.....	45
4.2.5 Arquitectura del hardware.....	47
4.3 Implementación de prototipos.....	50
4.3.1 Prototipo uno Alpha.....	50
4.3.2 Prototipo dos Beta.....	54
4.4 Validación del prototipo	61
4.4.1 Hardware	61
4.4.2 Visualización página web	62
4.4.3 Interacción del usuario.....	63
5. Conclusiones.....	65
6. Recomendaciones	66
Referencias Bibliográficas	68
Apéndices.....	70

Lista de Tablas

	Pág.
Tabla 1. Lista de requerimientos funcionales	37
Tabla 2 Lista de requerimientos no funcionales	38
Tabla 3. Validación de prototipo para el Hardware	62
Tabla 4. Validación de prototipo para la visualización de la página web.....	63
Tabla 5. Validación de prototipo para la interacción del usuario	64
Tabla 6. Registrar usuario	70
Tabla 7. Iniciar sesión	70
Tabla 8. Cerrar sesión	71
Tabla 9. Editar usuario	71
Tabla 10. Agregar plantas	71
Tabla 11. Modificar plantas	72
Tabla 12. Visualizar estado de plantas.....	72
Tabla 13. Medir datos de sensores	72
Tabla 14. Subir los datos de sensores a la nube	72
Tabla 15. Activar y desactivar luces	73
Tabla 16. Activar y desactivar ventiladores.....	73
Tabla 17. Activar y desactivar riego	73
Tabla 18. Registrar crecimiento de la planta	74
Tabla 19. Modificar crecimiento de la planta	74
Tabla 20. Eliminar crecimiento de la planta	74

Tabla 21. Almacenar el histórico de acciones autónomas	74
Tabla 22. Generar informes de medición de sensores	75
Tabla 23. Generar informe de crecimiento de las plantas.....	75
Tabla 24. Generar informe de histórico de acciones.....	75
Tabla 25. Permitir operaciones con varios usuarios al tiempo	76
Tabla 26. Diseñar interfaz gráfica de usuario	76
Tabla 27. Diseñar el circuito a utilizar.....	76
Tabla 28. Realizar el montaje del circuito	77
Tabla 29. Conseguir hosting para el sistema	77
Tabla 30. Conseguir dominio público para la aplicación	77
Tabla 31. Usar certificados SSL	78
Tabla 32. Configurar servidor FTP.....	78
Tabla 33. Utilizar el protocolo SSH.....	78
Tabla 34. Hacer despliegue del aplicativo en un dominio público.....	79
Tabla 35. Implementar el protocolo MQTT	79
Tabla 36. Conectar el hardware con la base de datos del servidor	79
Tabla 37. Actualizar base de datos servidor	80
Tabla 38 Ser compatible con diferentes navegadores.....	80

Lista de Figuras

	Pág.
Figura 1. Diagrama del desarrollo Iterativo y creciente.	18
Figura 2. Cronograma de desarrollo del proyecto usando la metodología SCRUM donde se permite corregir el sistema durante cada iteración hasta que se encuentre listo para entrega.	20
Figura 3. Diagrama de señal Análoga vs Señal Digital	27
Figura 4. Figura 2. Diagrama de un microcomputador.....	28
Figura 5. Esquema de bloques general de un microcontrolador.....	29
Figura 6. Microcontrolador ESP32, con sensores, una pantalla y una tarjeta SD para almacenar los datos.....	34
Figura 7. EDYN sistema tipo jardín que consta de un sensor conectado con WIFI para envío de dato y válvulas de riego que funcionan con paneles solares y baterías de Litio.....	35
Figura 8. Distribución de un Sistema de vegetación usando Machine Learning para mejora en la mano de obra campesina.....	36
Figura 9. Casos de uso de la aplicación completa de GardenStats	39
Figura 10. Casos de uso para gestionar perfil de usuarios.....	40
Figura 11. Casos de uso para gestión de plantas.....	41
Figura 12. Casos de uso para interacción interfaz web.....	42
Figura 13. Diagrama de actividades para el hardware	43
Figura 14. Diagrama de actividades para el software	44
Figura 15. Modelo relacional.....	45
Figura 16. Diagrama Arquitectura del software	47

Figura 17. Diagrama Arquitectura del hardware hecho en Fritzing	48
Figura 18. Diagrama Arquitectura del hardware hecho en Proteus	49
Figura 19. Hardware del sistema prototipo Alpha	51
Figura 20. Vista prototipo Alpha inicio	51
Figura 21. Vista prototipo Alpha administración de plantas	52
Figura 22. Vista prototipo Alpha administración de crecimiento de plantas.....	53
Figura 23. Vista del informe estadístico en el prototipo Alpha	54
Figura 24. Hardware del sistema prototipo Beta	55
Figura 25. Prototipo Beta página de inicio	56
Figura 26. Prototipo Beta página principal	57
Figura 27. Prototipo Beta administración de plantas	58
Figura 28. Prototipo Beta administración de crecimiento de plantas	59
Figura 29. Prototipo Beta informes estadísticos de la aplicación para mediciones de sensores y crecimiento de la planta	60
Figura 30. Prototipo Beta bitácora de cambio de estado de los actuadores.....	61

Lista de Apéndices

	pág.
Apéndice A. Requerimientos del sistema.	70
Apéndice B. Página web del proyecto desplegado.	80
Apéndice C. Repositorio GitHub.	81

Glosario

Actuador: dispositivo eléctrico o electrónico encargado de realizar una acción según una condición establecida en un microcontrolador.

Cable UTP: es un tipo de cable que se utiliza para conexión UTP entre dos dispositivos para transferencia de internet. También es utilizado para crear conexiones en protoboard.

Certificados SSL: secure Sockets Layers se trata de un protocolo cripto grafico que permite una conexión segura en la red.

Hosting: Servicio de almacenaje en un servidor que proporciona espacio para subir la base de datos, página web y otros archivos.

Protocolo FTP: file transfer protocol es un protocolo de transferencia de archivos entre sistemas conectados a la misma red. a definición inicia en minúscula.

Protocolo MQTT: machine queuing telemetry transport es un protocolo de transferencia de información entre maquinas que se encuentran en ubicaciones distinta, funciona por publicación y suscripción a través de un tópico.

Tópico en MQTT: cadena que lleva una estructura de clave-valor que al ser recibido por un controlador se encarga de transferir los valores a la base de datos según el identificador.

Resumen

Título: Sistema de control automatizado para pequeños cultivos enfocado en monitoreo y regulación.*

Autor: Alvaro Luis Rios Garzón y Cristian Guillermo Serrano Acevedo **

Palabras Clave: Cultivo, Sensor, Microcontrolador, Arduino, Temperatura, Luminosidad, Humedad, Planta.

Descripción: Este documento presenta un sistema automatizado de cultivos de fresa con el fin de obtener los mejores resultados en las cosechas recogidas mientras se intenta evitar enfermedades y bacterias que aparecen cuando no se tiene un control estable de sus condiciones ideales. El principal objetivo del proyecto fue crear un prototipo de invernadero capaz de monitorear el progreso del cultivo a través de una página web que permitirá visualizar los datos obtenidos por los sensores en diferentes horas del día. Para ello se utilizó un sensor de humedad y temperatura DHT11, cuatro sensores de humedad del suelo HL-69 y un sensor de luminosidad BH1750 para la luz. Por fines prácticos se usó un módulo Arduino ESP32, dicho microcontrolador, también se encargó de subir la información a la nube vía WI-FI. Finalmente evaluó el funcionamiento de las mediciones de los sensores y la toma de decisiones de acuerdo con los datos procesados a tiempo real.

* Trabajo de Grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: Lola Xiomara Bautista Roza. Doctor en Ciencias y Tecnologías de la comunicación y la Información. Codirector: Urbano Eliécer Gómez Prada Magíster en Ingeniería Área Informática y Ciencias de la Computación.

Abstract

Title: Automated control system for small crops focused on monitoring and regulation. *

Author(s): Alvaro Luis Rios Garzón and Cristian Guillermo Serrano Acevedo **

Key Words: Crop, Sensor, Microcontroller, Arduino, Temperature, Luminosity, Humidity, Plant.

Description: This document presents the development of an automate strawberry system in greenhouse mode with the purpose of getting better result in a strawberry crop and avoid sickness and bacteria who appear when there is no control in ideals conditions of this Plant. The main objective of this project was to create a greenhouse prototype that is to be able to get information about the crop state in a web page that shows data captured by sensors across different hours of the day. To achieve this, it was necessary to use a humidity and temperature sensor DHT11, four ground humidity sensors HL-69 and a photoresistor to measure luminosity. In this work was used an ESP32 Arduino module. A microcontroller that can send data to cloud via WI-FI. Finally, there was evaluated the performance of the sensors measurements and actions made by the system using the processed data in real time.

* Bachelor Thesis

** Faculty of Physical-Mechanical Engineering. Systems Engineering and Informatics Department. Advisor: Lola Xiomara Bautista Roza. Doctor en Ciencias y Tecnologías de la comunicación y la Información. Co-Advisor: Urbano Eliécer Gómez Prada Magíster en Ingeniería Área Informática y Ciencias de la Computación.

Introducción

Colombia posee unas excelentes condiciones para cultivos, puesto que en este existen hasta ocho tipos distintos de suelos (Instituto Geográfico Agustín Codazzi [IGAC], 2021). Sin embargo, mientras otros países hacen uso de tecnologías avanzadas, como el uso robots y drones, o incluso aplicaciones que permiten saber el estado en que se encuentran los cultivos.

En Colombia no han sido implementadas tecnologías en la agricultura a mayor escala. Incluso, cuando se habla de tecnología y de agricultura se pueden entender como dos temas separados, en donde uno no tiene relación con el otro, pero la realidad es que el uso de tecnologías en la agricultura cada día se vuelve más vigente, en donde el agricultor necesita encontrar la forma más eficiente de cuidar y asegurar la supervivencia de sus cultivos (Agrimanager, 2019).

Es pertinente que en Colombia se aumente el uso aplicaciones tecnológicas, ya que hay estudios que afirman que la implementación de tecnologías en la agricultura puede llegar a generar entre un 30% a 45% de mayor rentabilidad con respecto a cosechas más tradicionales (Anónimo, 2021).

Cabe destacar que la utilización de aplicaciones tecnológicas supondría un gran reto, sobre todo en términos de conectividad, ya que, a pesar de que en zonas urbanas es fácil acceder a conexiones 4G, existen zonas rurales en donde la gente sigue conectada a 3G o incluso 2G. Tigo tomó la iniciativa de encender más de 2300 antenas en la banda de 700Mhz, para así generar un impacto en zonas rurales (Anónimo, 2021). Para Tigo existen cuatro tecnologías clave para poder

aumentar la productividad en la agricultura: Los sensores remotos, los drones, los vehículos autónomos y los robots agrícolas.

Se plantea la automatización y monitoreo de un cultivo de fresas a pequeña escala, donde el usuario tendrá acceso a diferentes datos acerca de las plantas de fresa, como la humedad del suelo, la temperatura y la luminosidad con el uso de sensores y un microcontrolador para reunir esta información y gracias al uso de tecnologías libre de distribución se va a crear una página web que permitirá visualizar la información.

1. Descripción del proyecto

1.1 Planteamiento y justificación del problema

Según el IGAC, actualmente en Colombia, la ganadería, presente en más del 30% del país, solo podría implementarse en el 13,3%. El panorama agrícola es opuesto: el 4,7% tiene cultivos, cuando su área potencial es del 19,3% (Instituto Geográfico Agustín Codazzi [IGAC], 2021). Es decir, del suelo para producción agrícola casi todo está destinado a otras actividades, de tal forma que se descuidan los alimentos que se producen en el país.

Teniendo en cuenta la diversidad de flora, fauna y a su vez la posición geográfica que ha favorecido a Colombia con un clima tropical, el uso del suelo para la agricultura se ha visto limitado. Adicionalmente a ello, como se mencionó con anterioridad, el uso del suelo ha sido desaprovechado, puesto que en Colombia hay hasta 8 tipos distintos de suelos, pero no se utilizan los más adecuados para cultivar y se destina a otro tipo de actividades como mencionaba el IGAC.

Adicionalmente, cuando se trabaja con cultivos se pueden presentar problemas comunes que afectan su buen crecimiento. Los principales causantes de cultivos deficientes son factores como: problemas físicos relacionados al clima, suelo y agua; problemas del manejo del cultivo asociados al manejo de nutrientes, agua, etc. Y limitaciones bióticas como las plagas, malezas, entre otros (Chaudhary, Nanda, & Tran, 2003). Lo cual puede desembocar en una pérdida considerable de cultivos por parte de los campesinos.

En otros países ya han recurrido a otras tecnologías para poder abordar este tipo de problemas, como el uso de drones con cámaras de alta definición para poder examinar los cultivos e incluso fumigarlos. También se han usado otras tecnologías como los robots agrícolas para poder gestionar los recursos y los cultivos; y vehículos autónomos que reciben y toman ordenes en tiempo real. Una de las tecnologías más destacables es el uso de sensores remotos que permiten al usuario poder gestionar el estado en el que se encuentran las plantas, obteniendo datos sobre las condiciones en las que se encuentran como la humedad, la temperatura, salinidad y el estado del suelo; sin necesidad de ir físicamente a la ubicación de los cultivos (Anónimo, 2021).

Por lo tanto, conocer las condiciones ambientales en las que se encuentran los cultivos por medio del uso de sensores puede permitir trazar un plan de acción de tal forma que los resultados obtenidos en las cosechas agrícolas sean mejores, generando una mayor rentabilidad a los usuarios y minimizando las pérdidas.

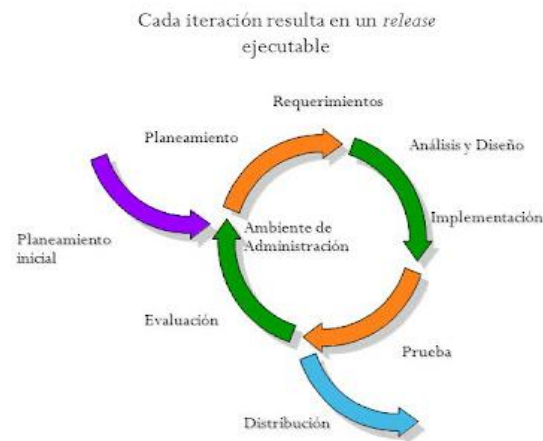
1.2 Metodología

Desde el punto de vista del desarrollo de software, se va a abordar una metodología de desarrollo iterativa, por su practicidad en el rápido avance y posibilidad de incorporar ajustes en etapas avanzadas de la construcción del software. Esta metodología de desarrollo aparece como una alternativa derivada de las debilidades que se encuentran en la metodología de desarrollo en cascada (Martinez, 2014).

El principio fundamental de la metodología de desarrollo iterativo es que consolida un pequeño conjunto de requerimientos en iteraciones, que se van desarrollando una después de otra hasta lograr que el conjunto de todas las iteraciones conforme la totalidad de los requerimientos del proyecto, haciendo que sea una metodología bastante utilizada junto a la programación extrema y SCRUM (Sabadí, 2017). la Figura 1 se puede apreciar el proceso que se realiza. Al igual que otras metodologías también se compone de varias etapas de desarrollo en cada iteración, iniciando cada una de ellas con la planificación de los requerimientos a incluir en la iteración y finalizando con la prueba y aprobación del sistema.

Figura 1.

Diagrama del desarrollo Iterativo y creciente.



Nota: Referencia (Zuppa, 2019)

La metodología desarrollo iterativo tiene algunas ventajas sobre la metodología de desarrollo en cascada y son:

- **Flexibilidad:** Los requerimientos no son fijos y pueden evolucionar durante el desarrollo del proyecto, y permite realizar ajustes de una manera flexible. Adicionalmente, el conocimiento que se adquiere en cada una de las iteraciones es muy útil para establecer un ajuste de los requerimientos para la siguiente iteración. Por otra parte, este conocimiento nos puede hacer reformar partes del sistema construidas en iteraciones anteriores.
- **Reducción en los riesgos:** Dado que las pruebas se realizan desde etapas tempranas en la construcción del proyecto, puede determinarse la viabilidad o eficiencia de las decisiones tomadas en cuanto al alcance y diseño de la solución, permitiendo que los elementos con más riesgo se desarrollen en las primeras iteraciones, con lo cual se pueda realizar una mitigación del riesgo en una etapa temprana que ayude a no comprometer los objetivos finales del proyecto.
- **Realimentación:** Contar con una versión preliminar de la aplicación, con un subconjunto de los requerimientos totales desde el mismo comienzo del proyecto, les permite a los usuarios que puedan revisarlos y proponer ajustes que consideren relevantes para mejorar su funcionalidad, e igualmente los desarrolladores pueden verificar rápidamente lo que funciona y lo que no, sin tener que esperar hasta el final del proyecto para realizar las modificaciones que se requieran.

Usando la metodología de desarrollo iterativo, en primera instancia, se definirán los requerimientos del proyecto, lo que permitirá recopilar información y definir la hoja de ruta del proyecto. A partir de esa información, se realizará un análisis y se diseñará el sistema. Cabe resaltar que el sistema necesita un hardware para recolección de datos, una base de datos para almacenarlos, una vez listo se puede definir la arquitectura de la página web.

Cuando el hardware y el software estén implementados, se pasará la etapa de pruebas donde se evaluará el desempeño del sistema. Con base en la evaluación, se determinará si el sistema está listo para su distribución o si este requiere ligeros cambios enfocados a mejorar su desempeño.

Este proceso se repetirá más de una vez si es necesario. A continuación, se puede apreciar un diagrama que representa dicho proceso en la

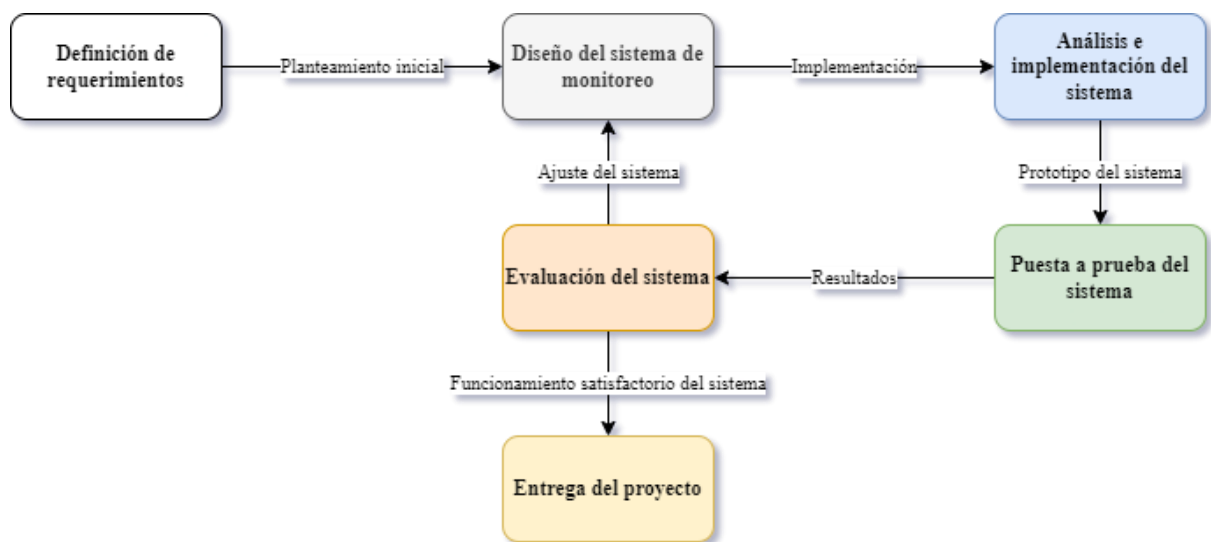
Figura 2.

Figura 2.

Cronograma de desarrollo del proyecto usando la metodología SCRUM donde se permite corregir el sistema durante cada iteración hasta que se encuentre listo para entrega.

Figura 2.

Cronograma de desarrollo del proyecto usando la metodología SCRUM donde se permite corregir el sistema durante cada iteración hasta que se encuentre listo para entrega.



2. Objetivos

2.1 Objetivo General

Diseñar un sistema automatizado de cultivos de fresa en modalidad invernadero, para el monitoreo y control de temperatura, luminosidad, humedad del ambiente y del suelo a través de tecnologías de libre distribución.

2.2 Objetivos Específicos

Definir el alcance de los requerimientos del sistema de control automatizado para el cultivo de plantas, el cual debe cumplir los siguientes requerimientos:

1. Recopilar y enviar datos de sensores vía WI-FI con el módulo ESP32 a una base de datos
2. Tomar decisión automática para el riego tomando como insumo los datos del monitoreo realizado a través de los sensores.
3. Permitir al usuario registrar el crecimiento de la planta.
4. Gestionar una bitácora con las acciones realizadas por el sistema según fecha y hora.
5. Generar informes estadísticos, tomados a partir de la información en la base de datos.

Diseñar la estructura del software utilizando diagramas UML y entidad relación.

Implementar la estructura del software tomando en cuenta los diagramas creados.

Realizar una prueba piloto para el sistema planteado, utilizando sensores accesibles en el mercado colombiano.

3. Marco de Referencia

3.1 Marco Teórico

Los primeros cultivos de plantas se dieron al final del periodo Paleolítico, la civilización dejó de recolectar alimentos de la naturaleza y se comenzó a utilizar la agricultura, con los cereales, la agricultura vino de la mano con el desarrollo de los asentamientos humanos (Vasey, 1992). Por lo tanto, la agricultura ha desempeñado un papel importante en la cultura humana, la cantidad de cultivos fueron creciendo poco a poco junto con las técnicas para obtener mejores resultados. En otras palabras, la necesidad de poder abastecerse fue lo que impulsó el uso de la agricultura en primer lugar.

Actualmente, la tecnificación de la agricultura ha hecho que se integren paulatinamente diversas máquinas y soluciones que permiten maximizar la productividad, disminuir costes de tiempo y materiales, e incrementar la calidad. Todos los cambios han ido en paralelo con el desarrollo de las industrias y tecnologías [4]. Todo con el fin de mejorar la eficiencia y satisfacer las necesidades del mercado. No obstante, el uso de la tecnología en la agricultura también se puede utilizar a nivel local, con el fin de obtener mejores resultados.

El uso del software de gestión agrícola resulta ser una herramienta fundamental en las empresas agrícolas modernas. Puesto que se pueden manejar grandes volúmenes de información que permiten mejorar la toma de decisiones (Calvo, 2019). Además, Colombia resulta ser una tierra rica, el uso de la tecnología resulta clave en la mejora de eficiencia de los cultivos agrícolas.

Teniendo en cuenta la eficiencia de la tecnología en los cultivos agrícolas, es muy importante tener una herramienta que permita a cualquier persona, almacenar y a su vez ayudar a tomar decisiones a lo largo de las producciones agrícolas y así obtener una mayor eficacia mientras se impulsa el uso de la tecnología en el país.

Ahora bien, hay que tener en cuenta cuales son las condiciones ideales para poder tener un cultivo eficiente de fresas en los invernaderos. Las fresas pueden llegar a soportar un rango amplio de temperaturas para ser una planta, no obstante, los mejores resultados de esta fruta se obtienen cuando se mantienen temperaturas desde los 12°C hasta los 25°C con una humedad de 60% y 80%. En cuanto a la técnica se recomienda usar goteo dado que otros métodos pueden causar exceso de humedad que permite la aparición de hongos (Agropinos, 2020).

Por otro lado, las plagas y enfermedades comunes de la planta de fresa están muy relacionadas a la humedad, las condiciones del suelo y la temperatura, por ello mantener las condiciones ideales ayuda a obtener una cosecha abundante y de buen tamaño, aunque en la práctica llevar a cabo esta labor puede resultar ser muy complicada.

Entre estas afecciones se encuentran tres plagas que son: Araña roja (*Tetranychus urticae*), Trips (*Frankliniella occidentalis*) y *Heliothis zea*, siendo de estas, la Araña roja la plaga más perjudicial para el cultivo de fresa puesto que si no se contiene puede matar la planta (Infoagro Systems, S.L., s.f). Aun así, las plagas en general son extremadamente peligrosas para los cultivos de plantas. Por lo tanto, su erradicación y prevención son fundamentales si se desea obtener buenos resultados.

En el caso de las enfermedades tenemos: Pobre dumbre gris (*Botrytis cinerea*), Oidio (*Sphaerotheca macularis* f. sp. *fragariae*), Mancha púrpura (*Mycosphaerella fragariae*), Bacterias (*Xanthomonas fragariae*), Antracnosis (*Colletotrichum* sp.) y hongos que pueden afectar tanto los cultivos como el fruto una vez cosechados. En el caso de los frutos cosechados una de las maneras de evitarlo es almacenando el fruto en bajas temperaturas lo más rápido posible (Infoagro Systems, S.L., s.f).

3.2 Marco Conceptual

A continuación, se presentan cada uno de los elementos correspondientes al marco conceptual.

3.2.1 Sensor

Un sensor es un dispositivo capaz de detectar magnitudes físicas o químicas a través de la energía del medio donde se mide, da una señal de salida traducible, la cual es la variable medida. Las magnitudes pueden ser, por ejemplo: intensidad lumínica, temperatura, distancia, aceleración, inclinación, presión, desplazamiento, fuerza, torsión, humedad, movimiento, pH, etc. (Pallás, 2003).

La importancia de los sensores trasciende a la vida cotidiana donde tienen una gran influencia en todos los objetos que rodean el medio, debido a que la mayoría de los dispositivos tecnológicos las utilizan para poder funcionar, algunos ejemplos son: las fotografías dado que las cámaras tienen un sensor (CCD o CMOS), sensor acústico (presión sonora) mejor conocido como micrófonos, sensor de proximidad en carros o celulares, entre otros. En otras palabras, la vida

como hoy la conocemos es posible gracias a los sensores. Cabe resaltar que muchas veces se utiliza transductor como sinónimo de sensor, pero hay ciertos factores que los diferencian.

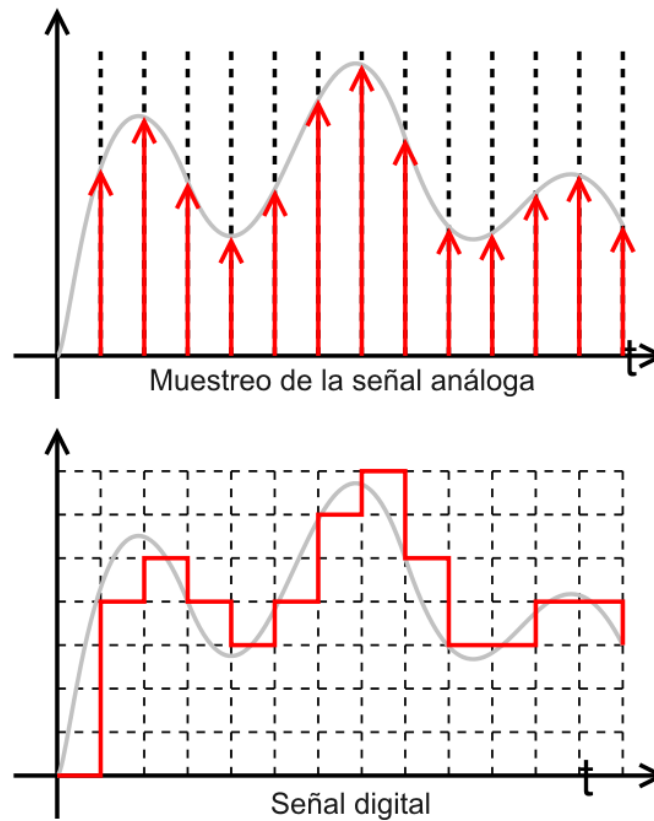
Los sensores se pueden clasificar de dos formas distintas, por energía se dividen en moduladores y generadores, y por la señal de salida en analógicos y digitales. A lo largo de este proyecto de investigación se utilizarán los últimos.

Los sensores analógicos obtienen los datos a través de una señal generada por algún fenómeno electromagnético, se representa como una función matemática continua. Es decir, son señales que presentan variaciones en su amplitud y periodo. Por otro lado, los sensores digitales son señales discretas que suelen estar representadas por sistema el binario 0 si es nivel bajo y 1 si es nivel alto, sus voltajes de operación suelen ser de 0 a 5V, no obstante, no se toman los valores intermedios. Por lo tanto, solo puede tomar dos valores posibles, a diferencia de las señales analógicas (Solectro, 2020).

Figura 3 En la Figura 3 se puede apreciar la diferencia entre los dos tipos de señales, en el caso de la analógica tiene una mayor gama de voltajes que pueden tomar y son usados normalmente en la electrónica de alta potencia, por otro lado, los digitales solo pueden tener dos estados y son utilizados en la electrónica de baja potencia.

Figura 3.

Diagrama de señal Análoga vs Señal Digital



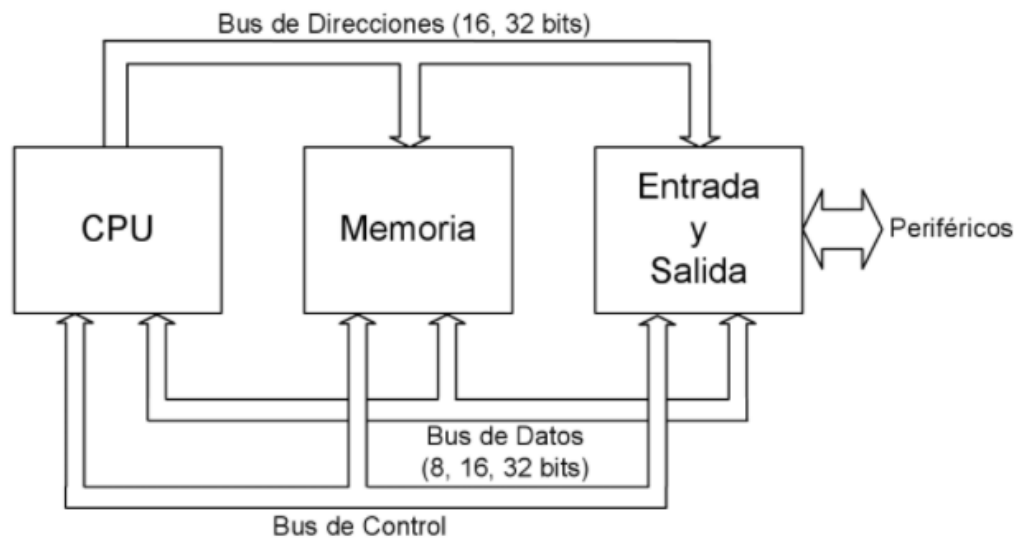
Nota: Referencia (Mantilla, 2013).

3.2.2 Microcontrolador

Antes de definir que es un microcontrolador, es pertinente revisar la definición de microcomputador y microprocesador. Los microcomputadores se dividen en tres componentes fundamentales que son: CPU (microprocesador), la memoria y entrada/salida. Los componentes se conectan a través de buses (líneas eléctricas). Los buses pueden transmitir direcciones de memoria, datos/instrucciones o señales de control (Valdés & Areny, 2007). A continuación, se puede apreciar el diagrama de un microcomputador.

Figura 4.

Diagrama de un microcomputador.



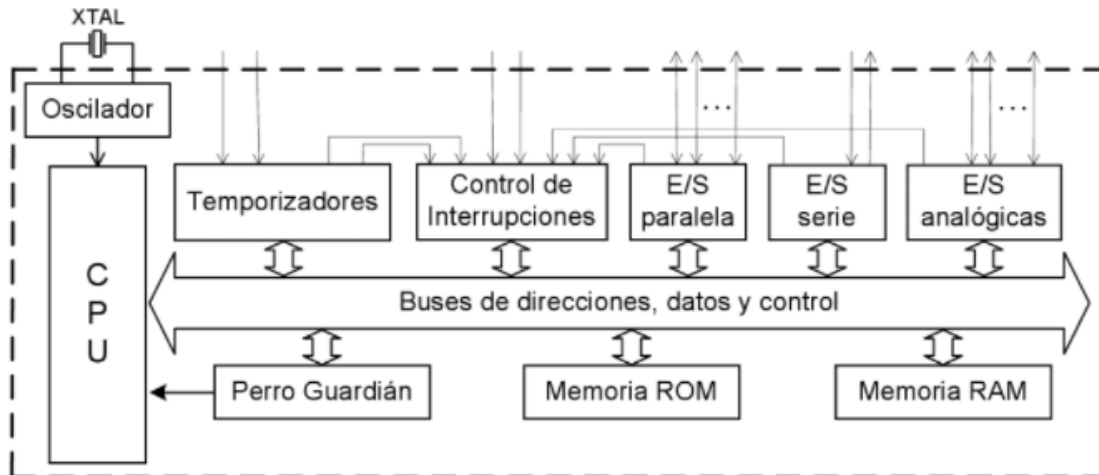
Nota: Referencia (Valdés & Areny, 2007).

Los microcontroladores son dispositivos capaces de realizar acciones o procesos programados por un usuario en lenguaje ensamblador y que son guardados en la memoria ROM (almacenamiento) para posteriormente ser ejecutados. Cabe resaltar que la memoria ROM es de solo lectura, por lo cual almacenará permanentemente las instrucciones programadas. A estas instrucciones se les conoce como procesos lógicos (Aguayo, 2004).

Un microcontrolador incluye en su interior las tres partes principales de un microcomputador. No obstante, cuenta con la particularidad de tener un oscilador que sincronizan las operaciones internas, en la siguiente figura se podrá apreciar la estructura interna de un microcontrolador.

Figura 5.

Esquema de bloques general de un microcontrolador.



Nota: Referencia (Valdés, 2007).

El perro guardián consta de un oscilador y un contador binario de N bits que se encargará de generar un reset (devuelve a la instrucción donde inicia el programa) en el microcontrolador una vez llega a contar N pulsos, para evitarlo el programador debe poner su contador en 0 muchas veces a lo largo del programa.

Para que un programa escrito en lenguaje ensamblador u otro lenguaje de microcontrolador pueda ser grabado en un microcontrolador, este debe ser decodificado en el sistema hexadecimal que es el sistema con el cual funciona los microcontroladores una vez están conectados al voltaje adecuado (Gridling & Weiss, 2007).

Los microcontroladores pueden ser reprogramables o no. A lo largo de este proyecto de investigación se utilizará una placa que consta de un microcontrolador reprogramable llamado ESP32.

3.2.3 Agricultura

La agricultura hace parte del motor económico de muchos países puesto que forma parte del sector primario de la economía, en otras palabras, se trata de un “Comoditie” o productos que pueden en el comercio o el sector financiero. La finalidad de la agricultura es la explotación del campo para proporcionar bienes, productos y materias primas que ayuden a mejorar la salud nutricional, la seguridad alimentaria, los procesos industriales y sobre todo la calidad de vida (Pineda, PRODUCTOS AGRICOLAS, s.f.).

Los cultivos agrícolas se pueden clasificar de diversas maneras, ya sea por el origen de las plantas (local u extranjera), la estación de siembra (primavera u otoño) o la finalidad del cultivo, para el ultimo según EOS (Earth Observing System [EOS], 2020) se pueden dividir en 6 grupos básicos:

3.2.3.1 Cultivos alimentarios

Sirven para satisfacer las necesidades alimenticias de las personas.

3.2.3.2 Cultivos para forraje

Es un cultivo especial para alimentar al ganado.

3.2.3.3 Cultivos textiles

Usado en la fabricación de ropa, productos del hogar, entre otros.

3.2.3.4 Cultivos oleaginosos

Se trata de aceites, ya sean industriales o comestibles.

3.2.3.5 Cultivos ornamentales

Son cultivos usados en la decoración del ambiente y el hogar.

3.2.3.6 Cultivos industriales

Utilizados para la manufacturación industrial como puede ser el caucho.

Por otro lado, la agricultura se puede clasificar según diversos factores como la dependencia de agua del cultivo (secano o regadío), la magnitud de la producción y su relación con el mercado (Subsistencia o industrial) o por su técnica/objetivo (Pineda, TIPOS DE AGRICULTURA, s.f.) que pueden ser:

- Agricultura tradicional es el más básico dado que se trata de cultivos con atrasó científico, que usan únicamente los recursos del lugar donde son plantados
- Agricultura moderna usa el conocimiento científico, técnico y tecnológico con el fin de obtener una mejor eficacia.
- Agricultura industrial al igual que la moderna utiliza tecnología, no obstante, está tiene la particularidad de aplicar costos medios de producción, con el fin de producir la mayor cantidad en el menor tiempo posible.
- Agricultura orgánica o ecológica se basa en preservar las características del entorno, es decir, que no resulte ser dañino para el medio ambiente y a su vez mejoren o mantengan la fertilidad del suelo.
- Agricultura por contrato que se trata de unos acuerdos entre agricultores y empresas, donde dichas empresas brindan sus servicios a los productores.

Finalmente se encuentra la clasificación de los cultivos agrícolas según el criterio utilitario, que agrupa los cultivos según el producto que genera, esta clasificación de especies es de las más populares, puesto que facilita la consolidación, intercambio y comparación de estadísticas agrícolas (Agrositio, 2019). Lo cual permite clasificar los cultivos de la siguiente forma:

- Cereales se caracterizan por tener ciclos de cultivo corto y sus granos poseen alto contenido de carbohidratos como el maíz y el arroz
- Leguminosas al igual que los cereales son cultivos de ciclo corto. no obstante, sus granos contienen alto nivel de proteínas como el frijol
- Oleaginosas sus cultivos tienen ciclos mayormente cortos, y se basan en la extracción del aceite de sus granos como la soya, el girasol para ciclo corto y largo la palmera de aceite
- Hortalizas también son de ciclo corto, del cual se comercializa alguno de sus órganos como el tomate, pimentón, entre otros.
- Frutales son de ciclo largos y como su nombre lo indica se trata de frutas que pueden ser usadas en las agroindustrias como el mango, aguacate.
- Ornamentales pueden tener los dos tipos de ciclos y se caracterizan por ser de carácter decorativo.
- Raíces y tubérculos tienen ciclos cortos, sus raíces y tallos tienen alto nivel de carbohidratos como la yuca, el ñame, etc.
- Cultivos para bebidas medicinales y aromáticas son cultivos que son utilizados para infusiones como el té, café, menta y tienen un ciclo variado de acuerdo con la planta.
- Cultivos tropicales son aquellos cultivos que suelen ser comunes en buenas condiciones tropicales como el café, cacao, caña de azúcar, entre otros.
- Pastos son cultivos de ciclo largo que son usados para alimentar rumiantes como la vaca a través del consumo de las hojas.

Cabe mencionar que hay diversos tipos de plantas que requieren cuidados especiales, ya sea trasplante, viveros, etc. Por ello se debe conocer a fondo que tipo de planta se desea cultivar y de esta manera determinar cuál es la forma más óptima de plantación y cuidado a lo largo de su crecimiento.

3.3 Antecedentes

Actualmente hay aplicaciones que utilizan de elementos tecnológicos aplicados a realizar un mejoramiento en diferentes aspectos de la actividad del sector agrícola. Una investigación muy remarcable consiste en el uso de la nano-biotecnología (NBT) donde se intenta implementar uso de nano-biosensores en la agricultura con el fin de obtener mejores mediciones (Singh, 2021). Lo cual permitiría cambiar la forma en la que se llevan a cabo los cultivos, incluyendo los industriales.

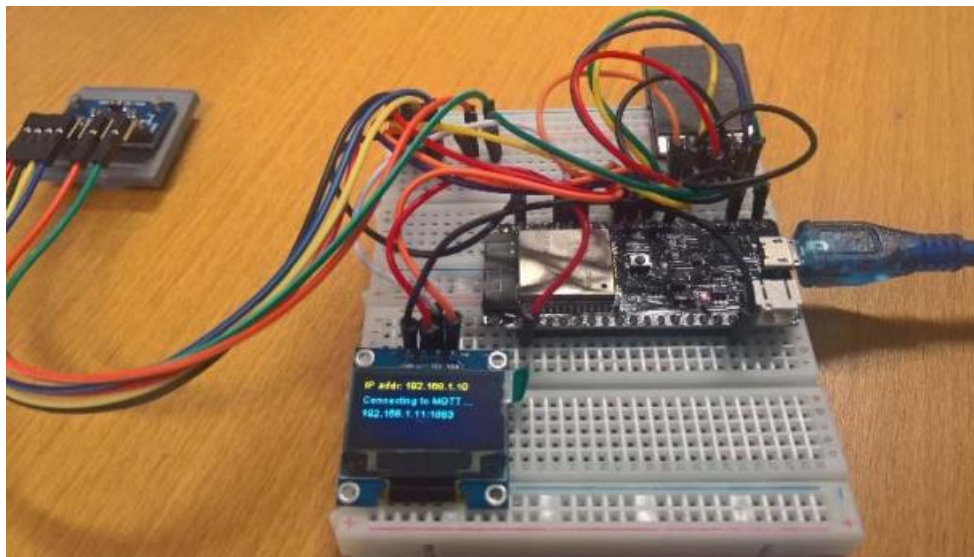
El uso de los biosensores en diversos campos como el ambiental, alimentos y por supuesto el sector agrícola ha sido un avance grande en la producción y resultados de las cosechas (Aharari, Yang, & Abe, 2021). El desarrollo de nanosensores permite elaborar proyectos que ocupen poco espacio, gasten pocos recursos energéticos y mejorar la precisión de tal manera que se pueda cambiar la perspectiva futura de las técnicas para la agricultura, y a su vez hallar nueva información en las plantas (Thirugnanasambandan, 2021).

La utilización de microcontroladores como es el caso del módulo ESP32 solucionan la problemática del envío de datos junto con el espacio y el bajo costo energético para el funcionamiento del sistema. Otra ventaja que posee el módulo ESP32 es su conectividad eficaz y segura requerida en los sistemas IOT modernos, A la comunicación y transferencia de datos inalámbricamente en dispositivos IOT se le conoce como las 6A dado que en ingles son: Anything, Anytime, Anyone, Anyplace, Any service, and Any network. (Li, Xu, & Zhao, 2015). En español: Cualquier cosa, cualquier hora, cualquier persona, cualquier lugar, cualquier servicio y cualquier red. En la Figura 6 se presenta un ejemplo de un sistema simple de lectura de sensores, con

impresión en pantalla, almacenamiento de datos local en una tarjeta SD y comunicación WI-FI. Este ejemplo es interesante dado que permite conocer cómo se puede desarrollar sistemas IOT usando el microcontrolador ESP32 y posteriormente formar sistemas más complejos.

Figura 6.

Microcontrolador ESP32, con sensores, una pantalla y una tarjeta SD para almacenar los datos.



Nota: Referencia (**Babiuch, Folynek, & Smutny, 2019**).

Un sistema similar al presentado en el documento es mostrado en la Figura 7 en la cual se utilizó un microcontrolador que se conectaba a una válvula con el objetivo de realizar riego automático de plantas de acuerdo con la medición de los sensores. A su vez, se podía observar la información de los sensores a través de una aplicación móvil y gracias al uso de paneles solares el consumo energético era bajo, de esta forma, el sistema puede mantener al usuario informado y al mismo tiempo cuidar la planta sin ser muy costoso. El conocer cómo se llevó a cabo el proyecto EDYN resultó ser muy útil en el desarrollo del prototipo de invernadero.

Figura 7.

EDYN sistema tipo jardín que consta de un sensor conectado con WIFI para envío de dato y válvulas de riego que funcionan con paneles solares y baterías de Litio.



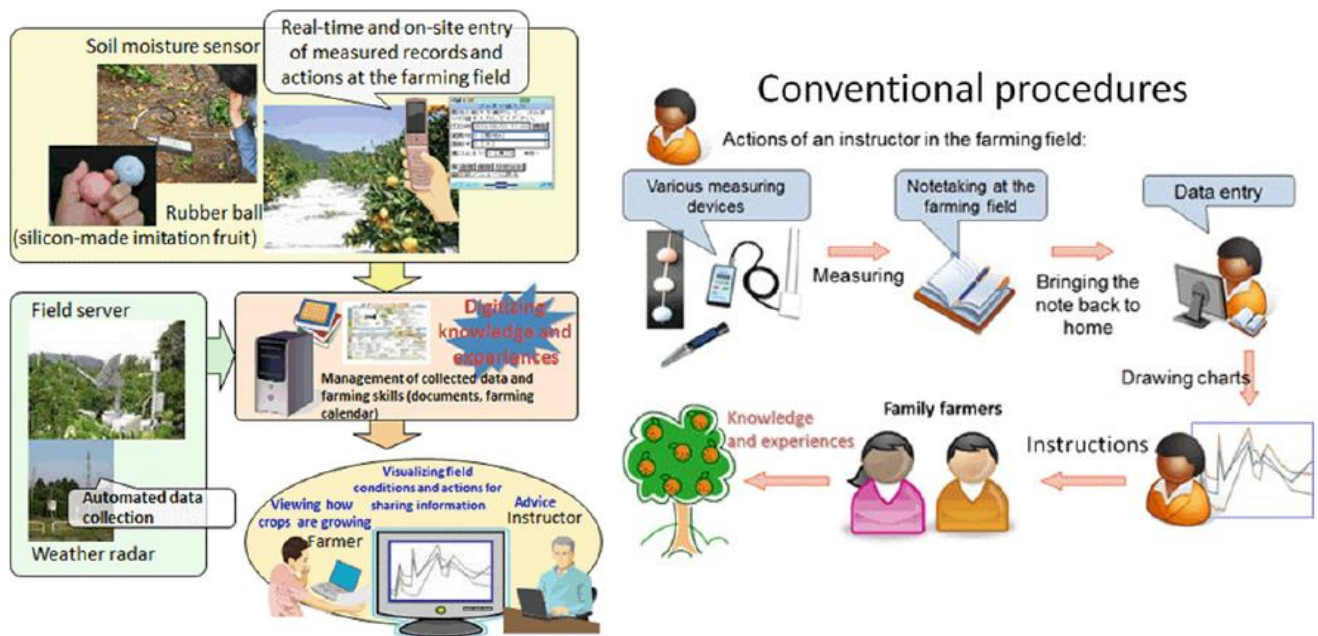
Nota: Referencia (Stinson, 2014).

Para el proyecto desarrollado a lo largo del documento es fundamental el uso de actuadores que van a realizar las acciones requeridas en el prototipo de invernadero, no obstante, la toma de decisiones utilizando inteligencia artificial puede ser un trabajo a futuro como se puede observar en el ejemplo de la Figura 8 se trató de un proyecto donde se utilizó machine learning con el fin de mejorar la toma de decisiones a partir de los datos recolectados y con ello incrementar la eficacia de las producciones agrícolas. Es pertinente aclarar que en el ejemplo las acciones eran realizadas

por familias granjeras. No obstante, también se podría utilizar machine learning para reducir la intervención humana mejorando la toma de decisiones autónomas del sistema a desarrollar en un futuro.

Figura 8.

Distribución de un Sistema de vegetación usando Machine Learning para mejora en la mano de obra campesina.



Nota: Referencia (Aharari, Yang, & Abe, 2021).

4. Resultados

A continuación, son presentados los resultados obtenidos en el proyecto siguiendo el orden los objetivos específicos. Es decir, los requerimientos, el diseño, la implementación y la validación. Para conocer un poco más acerca del proyecto en operación, puede visitar el siguiente enlace de [YouTube](#).

4.1 Requerimientos funcionales y no funcionales

4.1.1 *Requerimientos funcionales*

En la Tabla 1, se listan los requerimientos funcionales del sistema incluyendo el Hardware, se debe tener en cuenta que entre los requerimientos hay funciones que no requieren la intervención del usuario. Las especificaciones en detalle se pueden consultar en la sección de Apéndices.

Tabla 1.

Lista de requerimientos funcionales

Identificador (#RF)	Lista de requerimientos funcionales
RF01	Registrar usuario
RF02	Iniciar sesión
RF03	Cerrar sesión
RF04	Editar usuario
RF05	Agregar plantas
RF06	Modificar plantas
RF07	Visualizar estado de plantas
RF08	Eliminar plantas
RF09	Medir datos de sensores
RF10	Subir los datos de sensores a la nube
RF11	Activar y desactivar luces
RF12	Activar y desactivar ventiladores
RF13	Activar y desactivar riego
RF14	Registrar crecimiento de la planta
RF15	Modificar crecimiento de la planta
RF16	Eliminar crecimiento de la planta

Identificador (#RF)	Lista de requerimientos funcionales
RF17	Almacenar el histórico de acciones autónomas
RF18	Generar informe de medición de sensores
RF19	Generar informe de crecimiento de las plantas
RF20	Generar informe de histórico de acciones

4.1.2 *Requerimientos no funcionales*

A continuación, se presenta la Tabla 2 correspondiente a los requerimientos no funcionales del sistema, a diferencia de los requerimientos funcionales, los no funcionales representan características y/o restricciones del sistema. Para más detalle consulte la sección de Apéndices.

Tabla 2

Lista de requerimientos no funcionales

Identificador (#RNF)	Lista de requerimientos no funcionales
RNF01	Permitir operaciones con varios usuarios al tiempo
RNF02	Diseñar interfaz gráfica de usuario
RNF03	Diseñar el circuito a utilizar
RNF04	Realizar el montaje del circuito
RNF05	Conseguir hosting para el sistema
RNF06	Conseguir dominio público para la aplicación
RNF07	Usar certificados SSL
RNF08	Configurar servidor FTP
RNF09	Utilizar el protocolo SSH
RNF10	Hacer despliegue del aplicativo en un dominio público
RNF11	Implementar el protocolo MQTT
RNF12	Conectar el hardware con la base de datos del servidor
RNF13	Actualizar base de datos servidor
RNF14	Ser compatible con diferentes navegadores

4.2 **Diseño de prototipo**

Se presenta el diseño del prototipo hecho a partir de los casos de uso, diagrama de actividades, diagrama de clases, arquitectura del software y arquitectura del hardware.

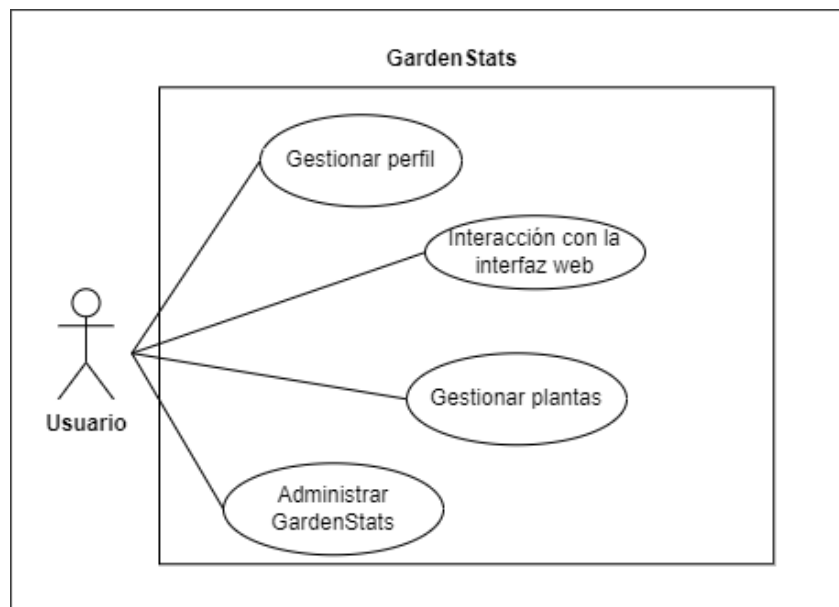
4.2.1 Diagramas de casos de uso

Los casos de uso representan las acciones que realizan los diferentes tipos de personas que acceden al aplicativo, en este caso se maneja un solo actor que es el usuario. Es decir, no hay roles adicionales como el de administrador que tengan funciones adicionales. Los casos de uso son presentados en diferentes diagramas para mayor claridad.

En la Figura 9 se presenta el diagrama de casos de uso general de GardenStats, por lo tanto, las acciones que realiza el usuario en la aplicación se desglosan en los casos de uso posteriores de tal manera que sea más fácil la comprensión del funcionamiento global del sistema. Para el caso de administrar GardenStats hace referencia al componente físico del proyecto que debe ser encendido u apagado por el usuario cuando vaya a realizar mediciones o por el contrario desactivarlas.

Figura 9.

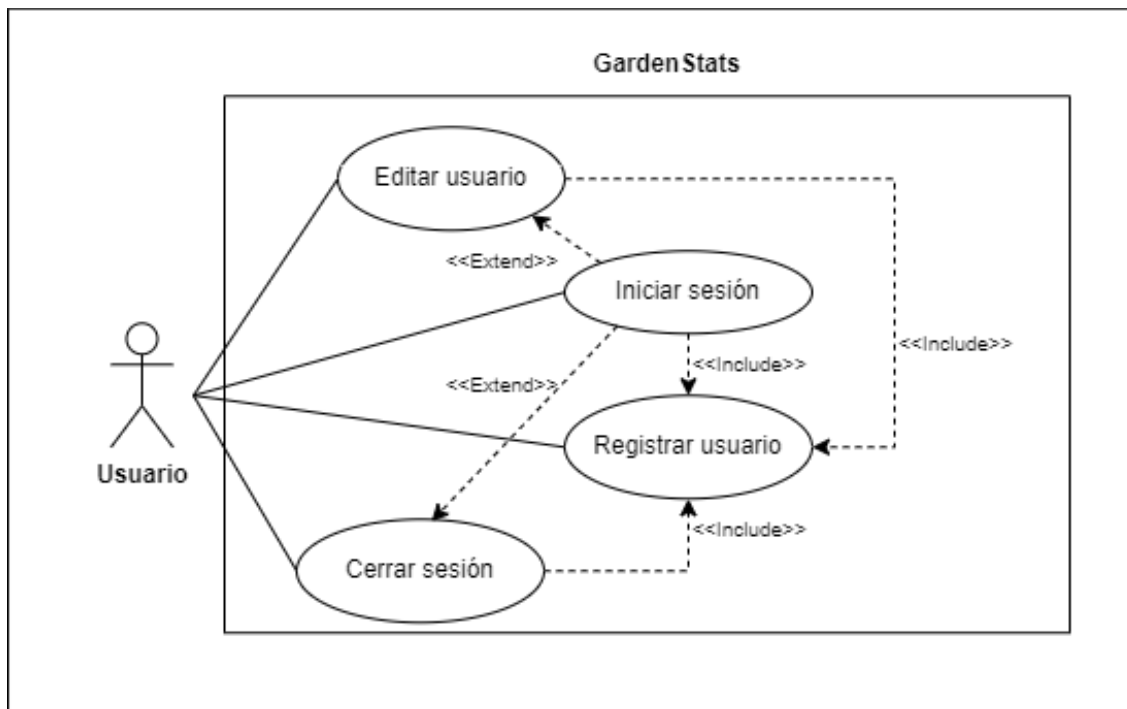
Casos de uso de la aplicación completa de GardenStats



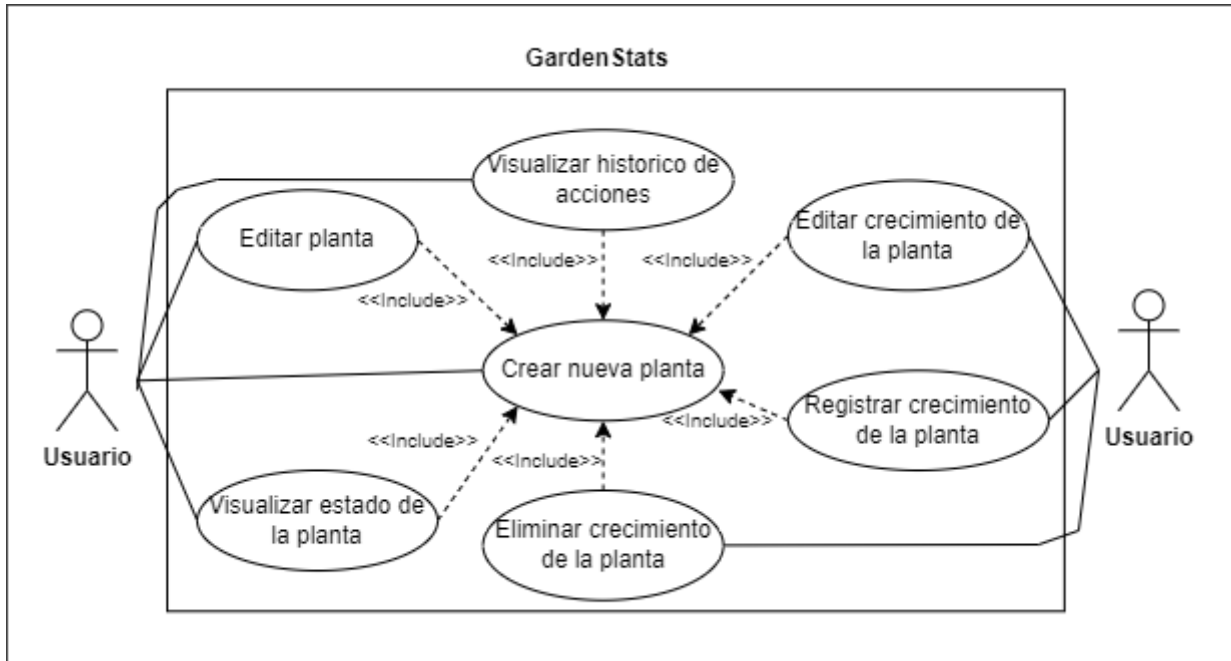
En la Figura 10 se encuentra el esquema de casos de uso para los usuarios donde todas las funciones que tienen los usuarios dependen de haber creado un usuario y en el caso de editar el usuario y cerrar sesión implica haber iniciado sesión en un principio.

Figura 10.

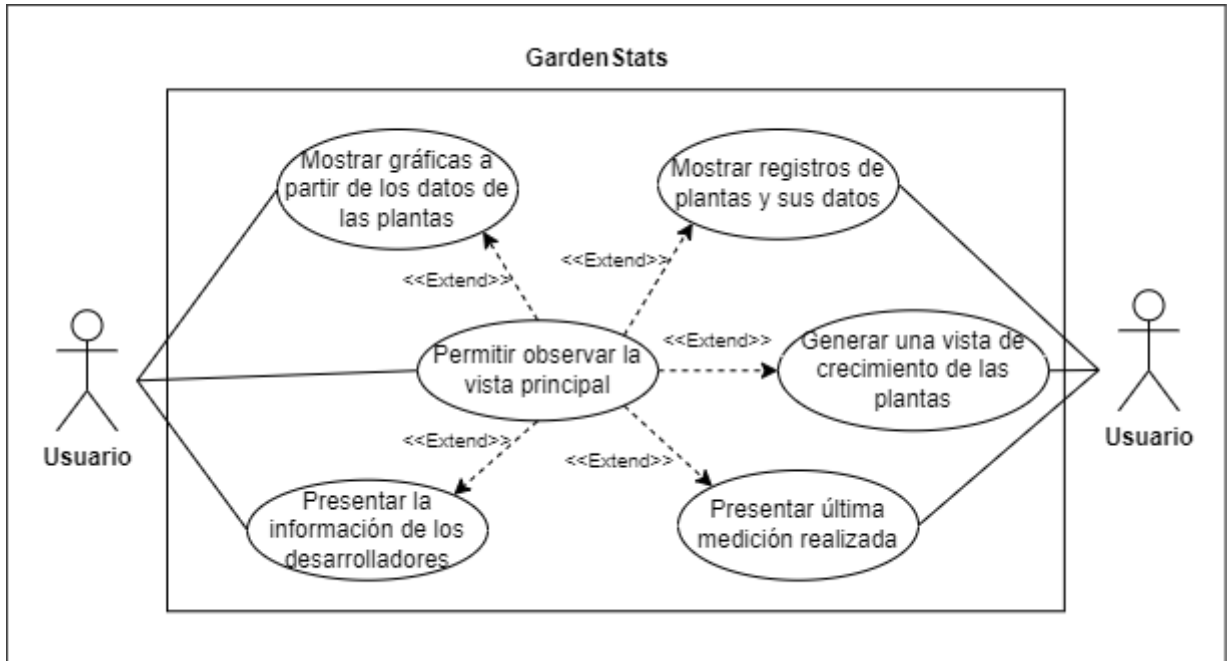
Casos de uso para gestionar perfil de usuarios



El diagrama de la Figura 11 se trata del caso de uso para la gestión de las plantas en el sistema, al igual que en el diagrama anterior solamente hay un actor. Para poder realizar las acciones con la planta se debe crear, la edición de la planta permite cambiar el nombre y también la eliminación de la planta junto con todos sus datos. El estado de la planta indica las mediciones que se han tomado de los sensores y si tiene luz, ventiladores o riego activo. Por otro lado, el crecimiento de la planta es digitado y modificado por el usuario cuando desee.

Figura 11.*Casos de uso para gestión de plantas*

Los casos de uso para la interacción de la interfaz web del sistema son representados en la Figura 12 al igual que en la figura anterior hay dos actores de usuario, para permitir una visualización más clara y sencilla del diagrama. Todas las opciones de interacción parten de poder observar la vista principal del aplicativo. El usuario podrá acceder a las diferentes funciones en un menú que separe y simplifique la información que sea requerida.

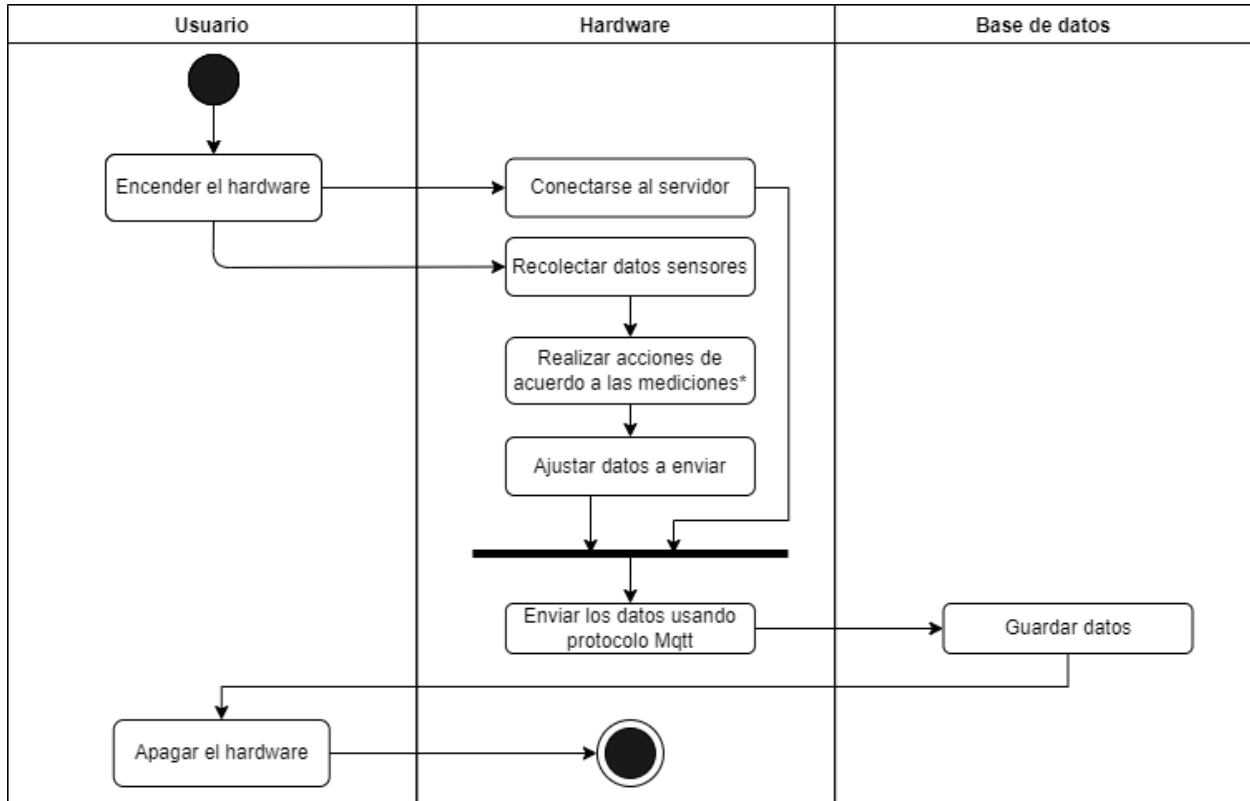
Figura 12.*Casos de uso para interacción interfaz web*

4.2.2 Diagramas de actividades

Los diagramas de actividades que se hicieron para representar el proceso de funcionamiento del sistema. Fueron divididos en varias secciones con el fin de facilitar el entendimiento de los diagramas. Adicional a ello, se realizó un diagrama que muestra el funcionamiento del Hardware que se puede apreciar en la Figura 13 como se puede observar, solo se requiere una acción por parte del usuario para que el sistema se encargue de guardar los datos enviados por los sensores. Las posibles acciones de acuerdo con los datos de los sensores pueden ser activar o desactivar los ventiladores, la motobomba y la luz. En el caso de haber cambiado el estado de alguna de estas acciones, entonces se reportarán estos cambios a la base de datos en un tópico distinto.

Figura 13.

Diagrama de actividades para el hardware

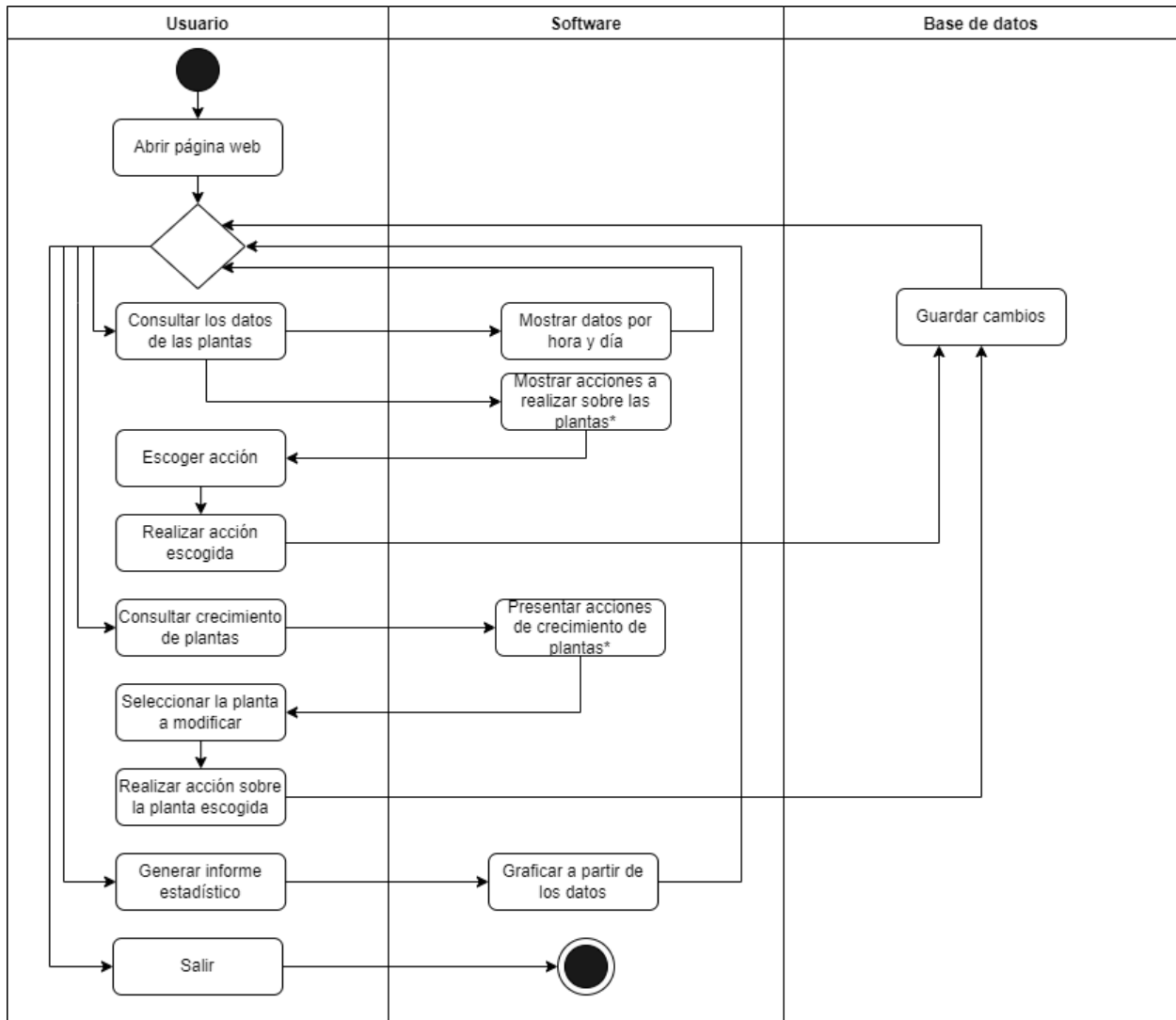


En el caso del software la primera parte del esquema es representado en la Figura 14 la cual está enfocada en el proceso que realiza el sistema cuando el usuario ingresa a la página y realiza las funciones que contiene la planta y su respectivo almacenamiento en la base de datos. En la página principal se puede acceder a diferentes pestañas que tienen sus caminos de posibles opciones, en el caso de las plantas cada cambio realizado por parte del usuario queda reportado en la base de datos. Las posibles acciones sobre las plantas serían crear, eliminar y editar la planta, lo mismo aplica para el crecimiento de la planta. Una vez seleccionada la acción a realizar y la planta

en la cual se quiere hacer la operación, se ajustará la base de datos de acuerdo con la solicitud enviada anteriormente.

Figura 14.

Diagrama de actividades para el software



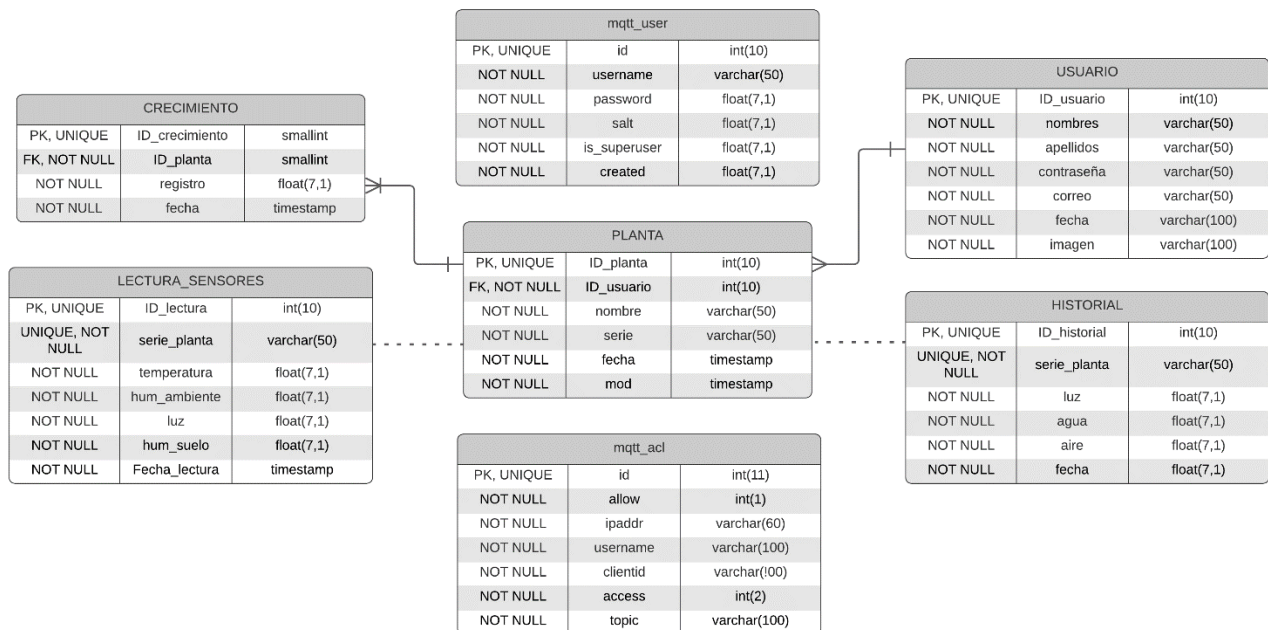
4.2.3 Modelo de la base de datos

El diagrama del modelo de la base de datos permite describir y representar la estructura de almacenamiento que tiene el sistema, ayuda a comprender los campos que se manejan en la base

de datos y su respectiva relación. En la Figura 15 se puede apreciar el modelo relacional de la base de datos que permite conocer como interactúa los datos recibidos por el módulo ESP32 con los campos que crea el usuario. La tabla de Mqtt tiene la función de validar las credenciales de envío de datos y de esta manera saber si permite la conexión. Las líneas punteadas representan un campo que no es llave primaria, pero debe tener el mismo valor, estos valores serán controlados por código. El campo serie de la tabla PLANTA se relaciona con línea punteada con los campos con serie_planta de las otras dos tablas.

Figura 15.

Modelo relacional



4.2.4 Arquitectura del software

El esquema se planeó usar MVC o modelo vista controlador que a su vez tendría una comunicación de datos maquina a máquina como cualquier proyecto IOT. El diagrama se puede apreciar en la Figura 16. En primer lugar, el entorno físico fue trabajado desde Arduino. Para él

envió de la información se utilizó el protocolo MQTT que permite tener una fiabilidad y seguridad en el cargue de datos a EMQX quien está encargado de reenviar los mensajes a quienes se encuentren suscritos al tópico o en otras palabras un token de conexión, NodeJs está suscrito al tópico y se encarga de la transferencia de datos a la base de datos MySQL ya sean enviados por la maquina o por el usuario.

Por otro lado, para mostrar la información de la base de datos en la página, se usó PHP quien mediante cadenas de texto plano permitió la transferencia de datos. Las cadenas de texto eran leídas por HTML5 y mostradas gráficamente al usuario. En el caso de la apariencia con el fin de presentar una apariencia más atractiva de la página web, se utilizaron diferentes librerías como Bootstrap, Font Awesome y Material Desing.

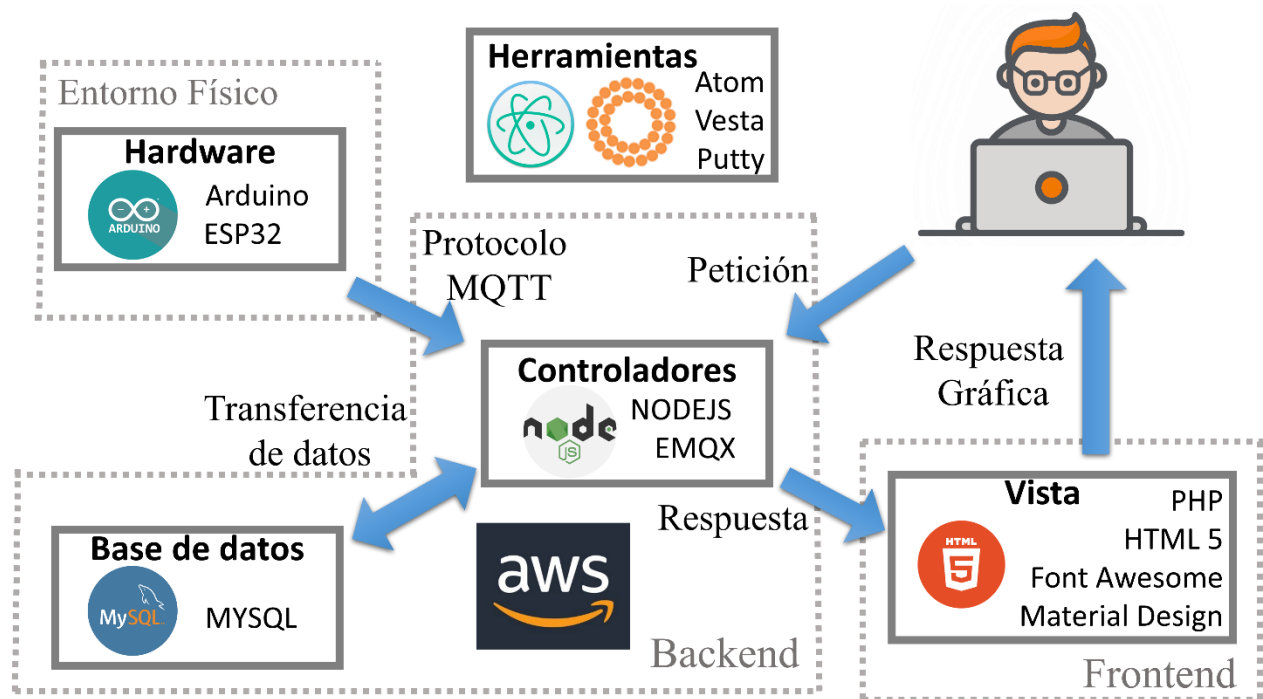
Cabe resaltar que toda la parte digital del software se montó en Amazon Web Service, para poder acceder a la consola de comandos de Linux fue requerido el uso Putty que permite hacer la conexión con el servidor mediante el protocolo SSH y a su vez configurar el protocolo MQTT y los certificados SSL de Vesta y EMQX. Putty también se utilizó para instalar NodeJS en el servidor. El uso de Vesta fue configurar MySQL, Apache, montar un cortafuegos adicional de tal manera que se tienen dos cortafuegos contando al propio de Amazon Web Service. En cuanto al entorno de desarrollo de código se usó Atom.

Para el dominio público se usó Hostinger que funciona como puerto a la ruta IP donde se encuentra alojada la página web y gracias a las DNS redirige al usuario a la ruta IP deseada, que en este caso fue la ruta donde se encontraba el servidor de Amazon Web Service. De esta manera

poder acceder fácilmente a la página web e igualmente tiene una opción para activar los certificados SSL para garantizar una conexión segura entre el usuario y el servidor.

Figura 16.

Diagrama Arquitectura del software



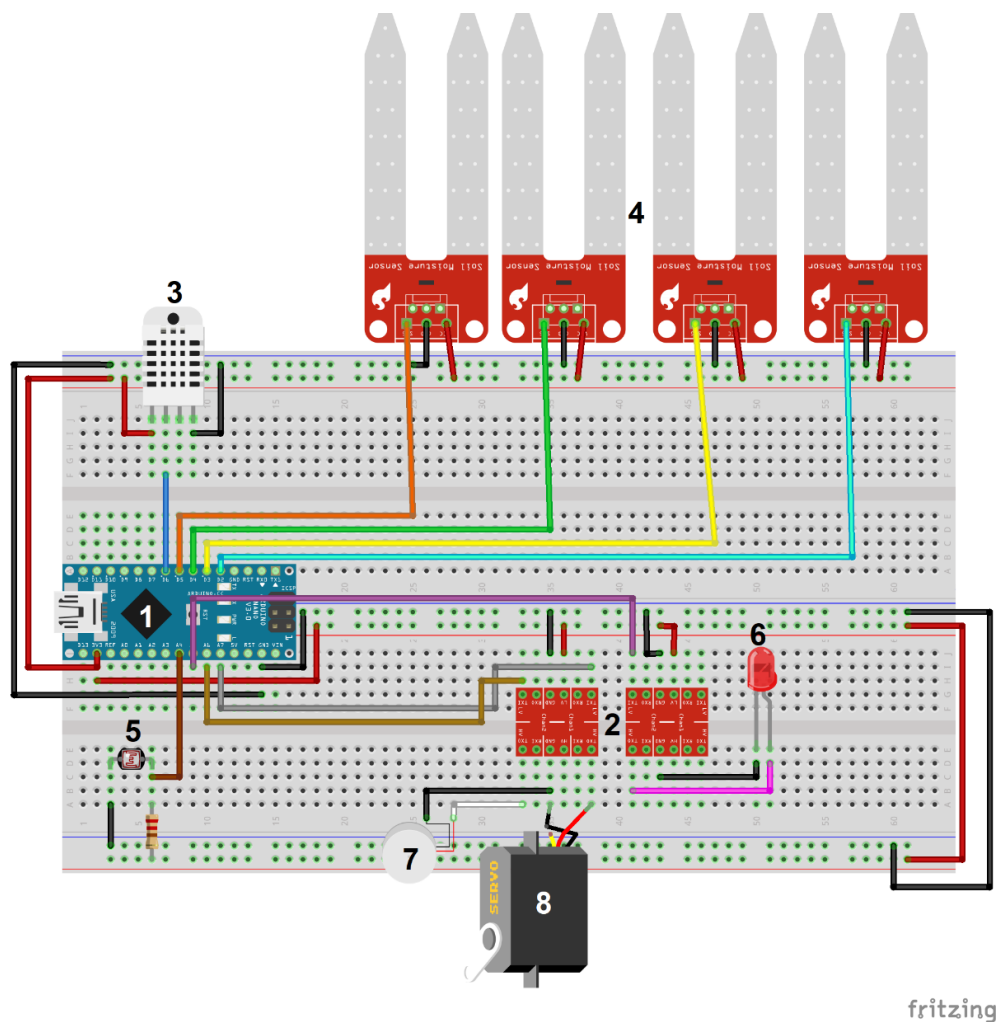
4.2.5 Arquitectura del hardware

La arquitectura del hardware fue hecha en dos softwares diferentes, para el primer caso en Fritzing permite visualizar de manera grafica e intuitiva las conexiones como se puede apreciar en Figura 17. La figura se puede dividir las secciones por números para saber a qué componente hace referencia. El primer componente representa el módulo Wi-Fi y Bluetooth EPS-32 que se encargará de hacer el puente entre la captura de los datos, su subida a la nube y el control de la motobomba, el foco y los ventiladores; el segundo es el convertidor lógico que sube el voltaje de 3.3 voltios a 5 voltios con el fin de usar Relés que se encargarán de informar a los actuadores cuando deben cumplir su función; el tercer componente es el sensor de humedad y temperatura

DHT-11; el cuarto representa los sensores de humedad del suelo HL-69; el quinto representa el sensor de luz BH1750, el sexto componente es el actuador del foco que se activa mediante el EPS-32; el séptimo, simboliza el actuador de los ventiladores y el octavo, hace referencia al actuador de la motobomba.

Figura 17.

Diagrama Arquitectura del hardware hecho en Fritzing

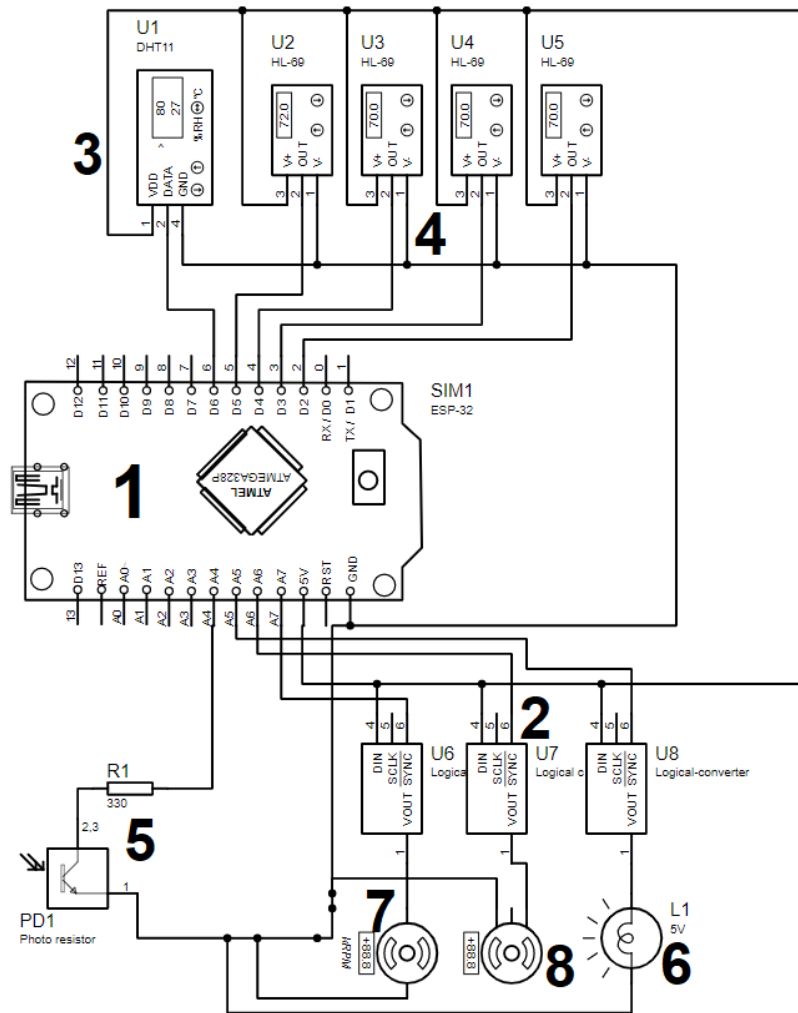


En la siguiente imagen se puede apreciar la Figura 18 que muestra a más detalles las conexiones a realizadas en el proyecto, fue diseñado con la aplicación de Proteus, al igual que en el diagrama de Fritzing se trata de una representación de los componentes utilizados y se maneja

la misma numeración para cada uno de estos. Cabe destacar que todas las conexiones del sistema fueron hechas con cable UTP y los sensores de humedad del suelo HL-69 se deben colocar sus puntas alrededor de las plantas de fresa en forma de cuadrado de tal manera que se pueda realizar la medición del área en la que se encuentra ubicado. Los datos proporcionados por el Hardware se reciben a tiempo real, no obstante, en la base de datos solo se almacenan cada cinco minutos.

Figura 18.

Diagrama Arquitectura del hardware hecho en Proteus



4.3 Implementación de prototipos

A lo largo del proyecto se desarrollaron dos prototipos los cuales fueron el Alpha y el Beta, los prototipos son explicados a continuación.

4.3.1 Prototipo uno Alpha

El prototipo Alpha fue la primera versión de GardenStats, tenía la particularidad de necesitar ayuda del usuario para la toma y cargue de datos de los sensores a la base de datos local. No obstante, contaba con gran parte de la funcionalidad final, como por ejemplo la visualización de gráficas que mostraban las mediciones de los sensores para las diferentes plantas a lo largo del tiempo. A continuación, se encuentra el siguiente enlace de [YouTube](#) donde se puede apreciar el funcionamiento de este prototipo.

En la Figura 19 se puede observar el montaje físico del prototipo Alpha, en esta versión usaba el Arduino R3 Uno, cuatro sensores de humedad del suelo HL-69, un sensor de humedad y temperatura DHT11, y un fotorresistor. En esta versión se capturaban los datos y eran recibidos en un archivo de Excel y desde allí el usuario debía hacer el cargue de los datos capturados a la base de datos local de MYSQL, además se debía especificar la planta a la cual se le realizaba la medición con su respectivo identificador, este dato era modificado desde el Excel antes de hacer la conversión del archivo en formato separado por comas CSV que permite una importación sencilla a la base de datos.

Figura 19.

Hardware del sistema prototipo Alpha



En la Figura 20 se puede apreciar la página de inicio de la aplicación donde podía acceder a las diferentes páginas y funciones que tenía el sistema a través del botón de menú en la esquina superior izquierda.

Figura 20.

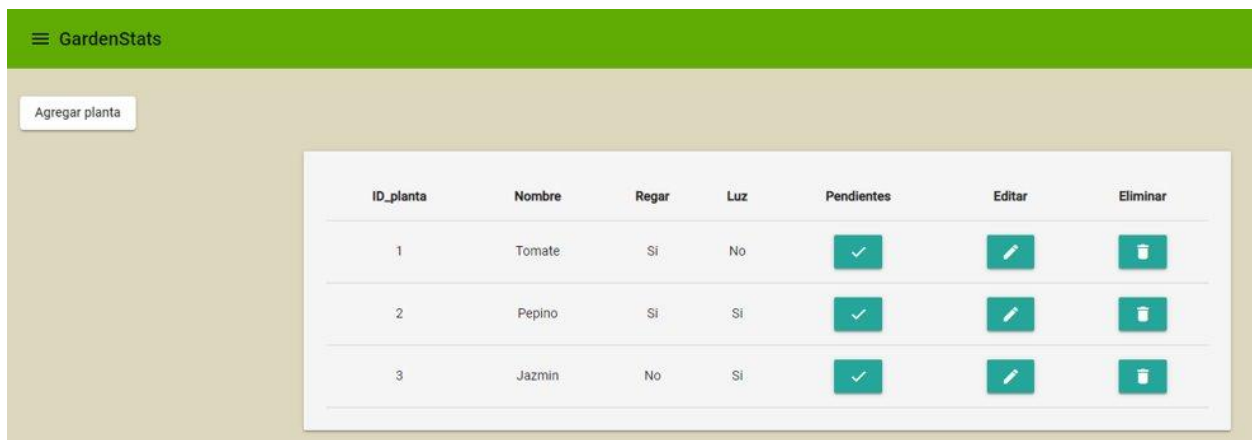
Vista prototipo Alpha inicio









En la página de las plantas que se puede apreciar la Figura 21 se podía visualizar el nombre de las plantas creadas y si era necesaria la exposición de luz y el riego. A su vez, permitía indicar si los pendientes de la planta habían sido completados, agregar nuevas plantas, editar el nombre de la planta y su eliminación cuando fuese considerado necesario.

Figura 21.

Vista prototipo Alpha administración de plantas



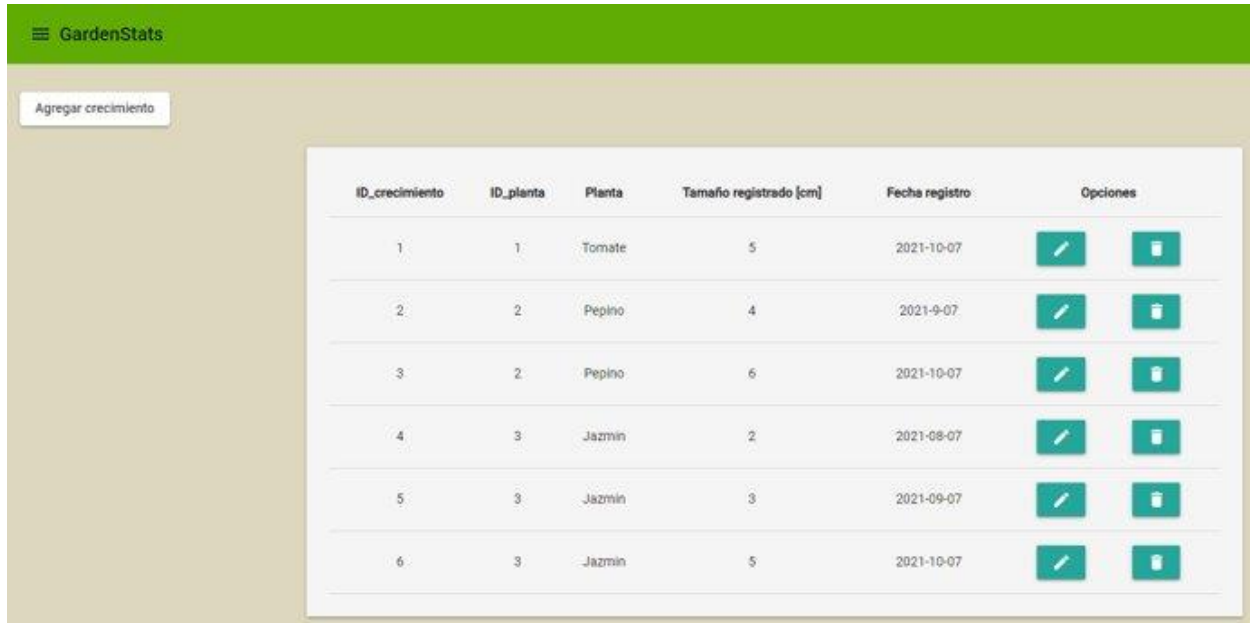
The screenshot shows the 'GardenStats' application interface. At the top left, there is a green header with the text 'GardenStats' and a hamburger menu icon. Below the header, there is a button labeled 'Agregar planta'. The main content area features a table with the following data:

ID_planta	Nombre	Regar	Luz	Pendientes	Editar	Eliminar
1	Tomate	Si	No	<input checked="" type="checkbox"/>		
2	Pepino	Si	Si	<input checked="" type="checkbox"/>		
3	Jazmin	No	Si	<input checked="" type="checkbox"/>		













En la Figura 22 se presenta la vista del crecimiento de plantas que es un control que el usuario digita según las mediciones del tamaño registrado en sus plantas en una fecha y así llevar más fácilmente un seguimiento del progreso del cultivo a lo largo del tiempo. Al igual que en el manejo de las plantas, el prototipo permitía la modificación y eliminación de las lecturas de crecimientos registrados con anterioridad.

Figura 22.

Vista prototipo Alpha administración de crecimiento de plantas



The screenshot shows the 'GardenStats' application interface. At the top, there is a green header with the text 'GardenStats' and a hamburger menu icon. Below the header, there is a button labeled 'Agregar crecimiento'. The main content area features a table with the following columns: 'ID_crecimiento', 'ID_planta', 'Planta', 'Tamaño registrado [cm]', 'Fecha registro', and 'Opciones'. The table contains six rows of data, each with a pencil icon and a trash can icon in the 'Opciones' column.

ID_crecimiento	ID_planta	Planta	Tamaño registrado [cm]	Fecha registro	Opciones
1	1	Tomate	5	2021-10-07	 
2	2	Pepino	4	2021-9-07	 
3	2	Pepino	6	2021-10-07	 
4	3	Jazmin	2	2021-08-07	 
5	3	Jazmin	3	2021-09-07	 
6	3	Jazmin	5	2021-10-07	 

La Figura 23 muestra la página de los informes estadísticos que permitía comparar las mediciones de temperatura, humedad del ambiente, humedad del suelo, luz y crecimiento a lo largo del tiempo, dicha grafica tenía la funcionalidad de ocultar plantas de tal manera que se podía comparar las plantas de interés en el caso de ser necesario. En este prototipo se tuvo que manejar un control de tiempos de mediciones para que las gráficas solo tomaran los valores cada seis horas. No obstante, se tenía que presionar el menú de gráficas para comparar cada valor de sensor lo que resultaba ser poco eficiente en el caso de querer observar todas las gráficas de las mediciones realizadas por el dispositivo.

Figura 23.

Vista del informe estadístico en el prototipo Alpha



4.3.2 Prototipo dos Beta

El prototipo Beta fue creado con el objetivo de evitar que el usuario realizará muchas acciones y ser capaz de realizar acciones autónomas. Para ello, según los rangos de datos de los sensores se activan o desactivan los actuadores quienes se encargan de las funciones como riego, ventilación y exposición a la luz y estos datos son enviados por un microcontrolador a través de Wi-Fi a la página web, donde es almacenado en la base de datos. A continuación, se puede observar un video con el funcionamiento del prototipo Beta en [YouTube](#).

En la Figura 24 se puede observar la estructura física del proyecto, a diferencia del prototipo anterior este cuenta con el módulo Wi-Fi y Bluetooth ESP32 para realizar el envío de los datos a la nube, los sensores utilizados fueron los mismos a excepción del fotorresistor que fue cambiado por un sensor de luminosidad BH1750. También cuenta con la incorporación de ventiladores, una motobomba y un foco. De acuerdo con las mediciones de los sensores el

microcontrolador emitía una señal de alto o bajo que era recibida por el conversor lógico, para avisarle al relé si debía activarse o desactivarse uno o más actuadores.

Figura 24.

Hardware del sistema prototipo Beta



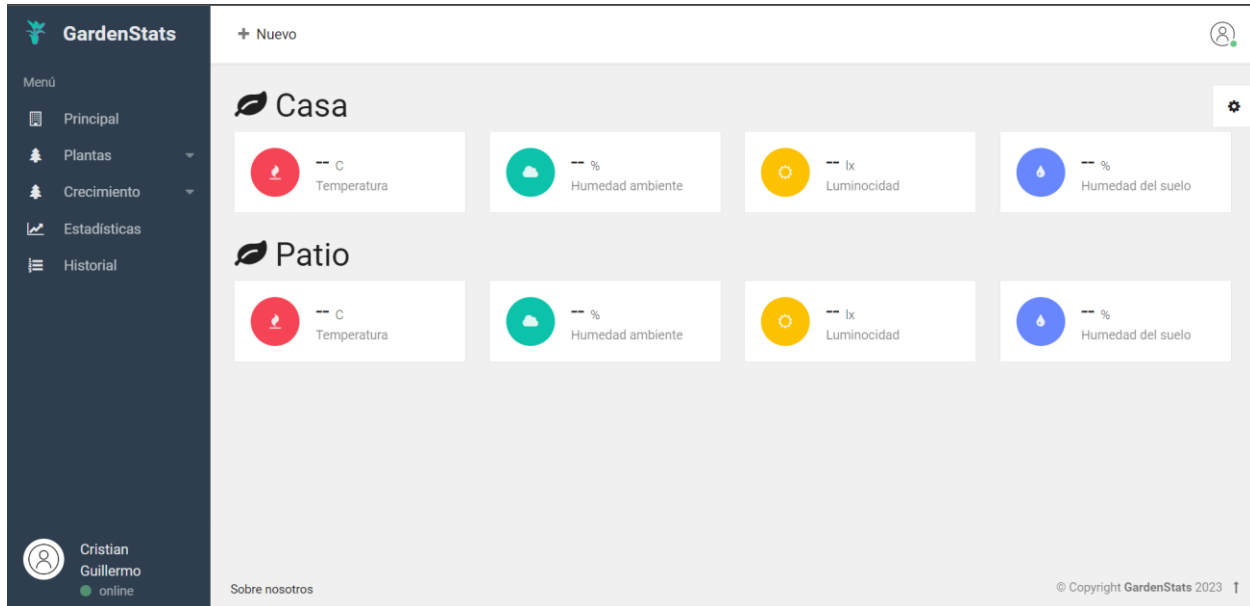
La página de inicio del prototipo Beta se puede observar en la Figura 25 con la nueva página se pueden crear usuarios y se actualizó el diseño con el fin de que solo los usuarios que han iniciado sesión a la página pudieran acceder a las otras vistas de la página como la administración de plantas, de crecimientos e informes estadísticos. Adicional a ello, se mantuvo la opción de ver el “acerca de” para conocer quiénes fueron los desarrolladores del proyecto.

Figura 25.

Prototipo Beta página de inicio



La vista de la página principal se puede observar en la Figura 26 está vista solo podrá ser accesible una vez se haya iniciado sesión. En ella se tiene la última medición registrada a tiempo real por el sistema de los sensores y el menú de opciones en la parte izquierda de la pantalla que redirigirán al usuario a las vistas que corresponden cada opción, adicional a ello también se puede ver el nombre de usuario al final de este menú. Cabe resaltar que la última medición de las plantas no tendrá valores en el caso de tener desconectado el circuito. Los datos de la última medición aparecerán en la página, aunque esto no significa que sean almacenados por la base de datos, dado que se programó una espera de 5 minutos por cada registro en la tabla.

Figura 26.*Prototipo Beta página principal*

En la Figura 27 se encuentra el manejo de las plantas, al igual que en el prototipo Alpha, permite la creación, modificación y eliminación de plantas. El manejo de la planta fue dividido en dos páginas, una que permite la creación y eliminación de la planta en la zona derecha de la vista y otra página para la edición. En la parte izquierda se presentan las plantas creadas por el usuario. Para la creación de plantas se debe especificar un número de serie de la planta que es único para cada planta y de esta manera distinguir a cuáles plantas se envían los datos de los sensores. Para hacer el cambio de planta a la cual enviar los datos se debe realizar por medio del código de Arduino.

Figura 27.*Prototipo Beta administración de plantas*

The screenshot shows the 'GardenStats' web application interface. On the left is a dark sidebar menu with options: Principal, Plantas (expanded), Agregar / Eliminar, Editar, Crecimiento, Estadísticas, and Historial. The user profile 'Cristian Guillermo' is shown as online. The main content area is titled 'Plantas' and contains a table with the following data:

name	Fecha de creación	Serie	Ultima modificación
Casa	2023-01-24 17:18:53	1234	2023-01-27 23:07:39
Patio	2023-01-27 22:50:51	5522	2023-01-27 22:50:51

Below the table are two forms. The 'Agregar planta' form has a title, a description, and two input fields: 'Nombre' (with example 'Ej: Casa Campo') and 'Serie' (with example 'Ej: 777222'), followed by a 'Registrar' button. The 'Eliminar planta' form has a title, a description, a dropdown menu showing 'Casa<->1234', and a red 'Eliminar' button.

La Figura 28 muestra el crecimiento de la planta, al igual que en el prototipo Alpha se mantiene gran parte de la misma funcionalidad, para que el usuario realice las mediciones al crecimiento del cultivo de fresas a lo largo del tiempo que considere oportuno o necesario. De esta manera hacerse una idea de cuánto tiempo puede tardar la finalización del cultivo. En este prototipo se anexó una validación adicional que solo permite modificar y/o eliminar el último crecimiento de la planta registrado por el usuario. Las mediciones permiten valores decimales por lo tanto es posible medir la planta con mayor precisión.

Figura 28.

Prototipo Beta administración de crecimiento de plantas

The screenshot shows the GardenStats web application interface. On the left is a dark sidebar menu with options: Principal, Plantas, Crecimiento, Agregar / Eliminar, Editar, Estadísticas, and Historial. The main content area is divided into two sections: 'Casa' and 'Patio'. Each section contains a table of growth records with columns for 'Registro (cm)' and 'Fecha'. The 'Casa' table has two rows: (5.0, 2023-01-27) and (4.0, 2023-01-27). The 'Patio' table has three rows: (2.0, 2023-01-23), (4.0, 2023-01-25), and (6.0, 2023-01-29). On the right, there are two forms: 'Agregar crecimiento' with a dropdown for 'Casa-->1234', a text input for 'Ej: 6cm', a date picker for '2023/05/09', and an 'Añadir' button; and 'Eliminar crecimiento' with a 'Seleccionar planta' dropdown.

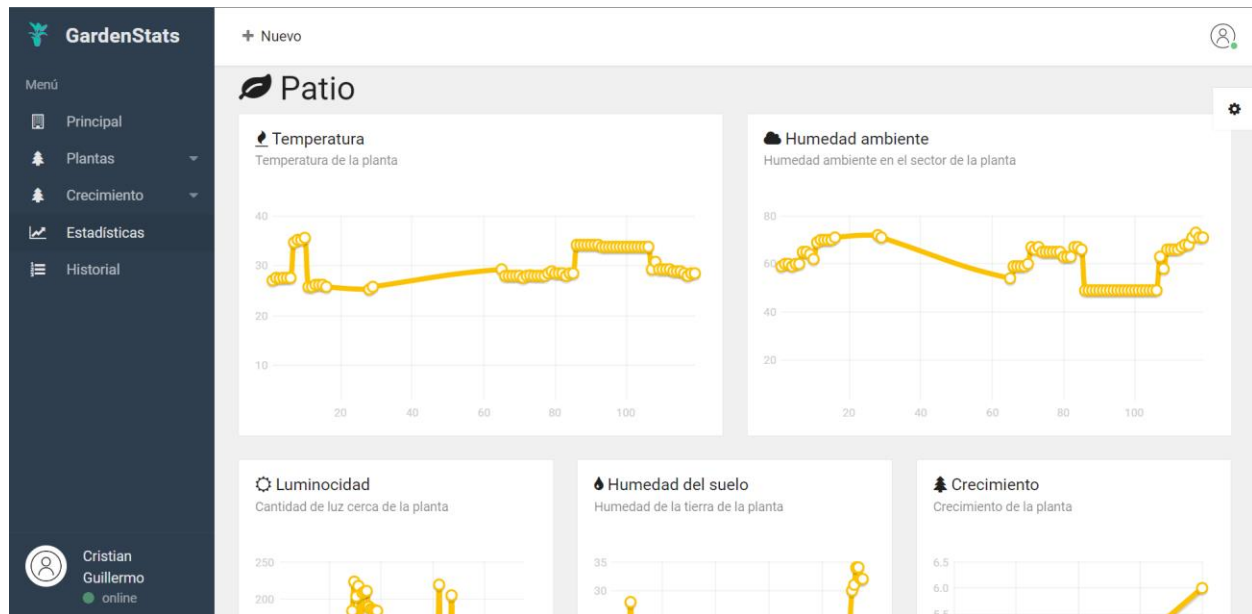
Registro (cm)	Fecha
5.0	2023-01-27
4.0	2023-01-27

Registro (cm)	Fecha
2.0	2023-01-23
4.0	2023-01-25
6.0	2023-01-29

La página que genera los informes estadísticos se llama estadísticas la cual es mostrada en la Figura 29 el prototipo Beta se caracteriza por generar graficas para las mediciones realizadas por los sensores y el histórico del crecimiento de la planta. En el caso de no tener valores registrados en alguna planta, la gráfica se verá vacía. A diferencia del prototipo Alpha, en este prototipo muestra el histórico de valores almacenados en la base de datos sin importar las horas de mediciones. Esto permite llevar un control más constante de las plantas dado el tiempo de guardado entre cada medición.

Figura 29.

Prototipo Beta informes estadísticos de la aplicación para mediciones de sensores y crecimiento de la planta



La página de historial presenta la bitácora de las acciones realizadas de forma autónoma por el prototipo como se puede observar en la Figura 30. Esta bitácora indica el día hora y minuto en el que algún actuador cambia de estado y se presenta en forma de tabla. Cuando una planta no ha realizado mediciones, la tabla debe mostrar un registro en el que todos los actuadores se encuentren en estado apagado. Se decidió usar esta forma de presentar los datos para simplificar la comparación entre los cambios de estado de cada actuador del sistema. Es pertinente recalcar que el sistema compara el último estado registrado para la planta que haya sido seleccionada y que en el caso de que no haya cambios, NodeJS se encarga de omitir el valor y solo registrar en la base de datos los cambios.

Figura 30.

Prototipo Beta bitácora de cambio de estado de los actuadores

The screenshot shows the GardenStats web application interface. On the left is a dark sidebar menu with options: Principal, Plantas, Crecimiento, Estadísticas, and Historial. At the bottom of the sidebar, the user profile 'Cristian Guillermo' is shown as 'online'. The main content area has a '+ Nuevo' button and a search icon. It displays two tables: 'Casa' and 'Patio'. Each table has columns for 'Bombillo', 'Bomba', 'Ventilador', and 'Fecha de registro'. The 'Casa' table shows one row with all actuators 'Apagado' and a timestamp of 2023-01-29 15:36:28. The 'Patio' table shows five rows with various states: 'Apagado', 'Encendido', and 'Apagado' for the actuators, and timestamps ranging from 2023-01-28 12:25:45 to 2023-01-29 17:30:39.

Casa			
Bombillo	Bomba	Ventilador	Fecha de registro
Apagado	Apagado	Apagado	2023-01-29 15:36:28

Patio			
Bombillo	Bomba	Ventilador	Fecha de registro
Apagado	Apagado	Apagado	2023-01-29 15:36:41
Apagado	Encendido	Apagado	2023-01-29 17:25:09
Apagado	Apagado	Apagado	2023-01-29 17:30:39
Apagado	Apagado	Encendido	2023-01-28 12:25:45

4.4 Validación del prototipo

Para validar el prototipo Beta, se tuvo en cuenta el siguiente plan de pruebas donde los desarrolladores del proyecto validaron la información los diferentes componentes del sistema. El funcionamiento del hardware, la visualización de la página web y la interacción del usuario con los componentes de la página.

4.4.1 Hardware

En la Tabla 3 se puede apreciar la validación que se le realizó al Hardware. Registra los datos obtenidos a través de los sensores de humedad, temperatura y luminosidad. Se conecta al servidor para hacer el cargue de los datos a través de Wi-Fi. Por otro lado, de acuerdo con las condiciones ambientales de la planta de fresa activa el riego, ventiladores y luz.

Tabla 3.*Validación de prototipo para el Hardware*

#	Usuario	Sistema	Datos	Revisar
1	Encender GardenStats	Lectura de sensores		Que los datos sean almacenados en las variables y que solo presenten valores numéricos
2		Conectarse al servidor donde se encuentra alojada la página		Que se haya logrado una conexión satisfactoria con el servidor
3		Enviar los datos a través de Wi-Fi y almacenarlos en la base de datos de la nube		Que los datos hayan quedado correctamente indexados en las tablas. Lectura de sensores e Historial.
4		Activar acciones autónomas		Que se active y/o desactiven los actuadores que permite controlar el estado de la planta. Riego, ventiladores y luz.
5	Apagar GardenStats	Desconectarse del servidor y detener la toma de datos y actuadores		El servidor debe liberar correctamente la conexión con el ESP32, asegurándose que se no presenten bloqueos en la base de datos ni se reciban datos erróneos.
#	Errores – Descripción		Corregido	Fecha de Corrección

4.4.2 Visualización página web

En el caso de la Tabla 4 presenta la validación de las acciones que debe hacer el sistema en el apartado visual de la página, como se mencionó con anterioridad el único actor es el usuario. Las acciones se dividieron por cada página que posee la aplicación.

Tabla 4.*Validación de prototipo para la visualización de la página web*

#	Usuario	Software	Datos	Revisar
1	Ingresar a la página WEB	Cargar texto e imágenes y mostrar botones		La página web se puede abrir sin problemas y la apariencia con sus datos se puede visualizar, y la página debe contar con diseño adaptativo.
2	Entrar a la página de plantas	Mostrar datos básicos de la planta y también los registrados por los sensores		Los datos son imprimidos correctamente en la página web y validar la interdependencia de los datos tanto para las plantas como los usuarios.
3	Entrar al crecimiento de la planta	Mostrar los datos del crecimiento registrado para cada planta si existen		El software debe mostrar el histórico de crecimientos registrado por los usuarios para cada planta.
4	Generar informe estadístico	Mostrar grafica con los datos sobre la planta especificada de acuerdo con el crecimiento, datos de sensores u acciones autónomas		La grafica es creada con los datos almacenados y se puede visualizar en la página según el tipo de informe que sea solicitado.
5		Imprimir acciones automáticas de los actuadores.		El software debe mostrar los cambios de estado de los actuadores de acuerdo con las mediciones realizadas por los sensores.
#	Errores – Descripción		Corregido	Fecha de Corrección

4.4.3 Interacción del usuario

En el caso de la interacción del usuario se plasmó en la Tabla 5 su validación, donde se indica cuáles son las respuestas que debe realizar la aplicación de acuerdo con los parámetros u acciones hechas por el usuario. El usuario podrá agregar, modificar y eliminar la información de las plantas.

Tabla 5.*Validación de prototipo para la interacción del usuario*

	Usuario	Software	Datos	Revisar
1	Presionar un botón	Realizar la acción programada para dicho botón		Que funcione la acción asociada al botón.
2	Agregar planta	Debe solicitar el nombre de la nueva planta	Nombre: Fresa	Que no se puedan crear plantas con valor NULL o vacío.
3	Agregar crecimiento	Debe mostrar desplegable de las plantas para que el usuario escoja a cuál se va a asociar dicho crecimiento	-Planta: Fresa - registro: 5 (cm)	Se puede agregar un nuevo tamaño de planta seleccionada y se puede reflejar en el informe del crecimiento de la planta.
4	Modificar planta	Mostrar desplegable de las plantas y que el usuario seleccione cual va a modificar	-Planta: Fresa1 -nuevo nombre: Fresa Annapolis	Que la planta seleccionada se modifique el nombre indicado por el usuario
5	Modificar crecimiento	Mostrar desplegable de las plantas y que el usuario seleccione el último crecimiento que desea modificar	-Planta: Fresa -nueva medida: 6 (cm)	Que la planta seleccionada se modifique la medición del último crecimiento registrado por el usuario si existe, en caso contrario no debe aparecer en el desplegable.
6	Eliminar planta	Mostrar desplegable de las plantas, el usuario puede seleccionar cual planta eliminar	-Planta: Fresa	La planta especificada es eliminada con todos sus datos relacionados.
7	Eliminar crecimiento	Mostrar desplegable de las plantas y el usuario escoge cual planta eliminará su último registro de crecimiento	-Planta: Fresa Annapolis	El último registro de la planta seleccionada es eliminado, si no hay más registros, no debe aparecer la planta en el desplegable.
8	Generar informe estadístico	Despliega la lista posible de informes que pueden ser generados con los datos actuales del aplicativo	Informe: Histórico de crecimiento de plantas	Imprime en página los datos asociados al informe seleccionado, si un informe no es posible de generar en el momento la gráfica debe estar vacía.
#	Errores – Descripción		Verificado	Fecha de Corrección

5. Conclusiones

A lo largo de este proyecto se puede concluir que:

- Se cumplieron satisfactoriamente los objetivos planteados. Se diseñó un prototipo tipo invernadero para el cultivo de planta de fresa con acciones automatizadas utilizando tecnologías de libre distribución como fueron: Arduino, NodeJS, HTML5, MySQL, HeidiSql, Vesta, Putty, Atom, Bootstrap, Php, EMQX Font Awesome y Material Desing.
- La comunicación del módulo ESP32 con el servidor fue exitosa, desde el punto de conexión al servidor como envió de datos de forma segura usando el protocolo MQTT y fueron almacenados correctamente los registros en la base de datos.
- La toma de decisiones de forma autónoma permitió llevar el control de la planta de fresa disminuyendo la intervención humana y a su vez, simular como podría trabajar el sistema si se desarrollara el sistema a gran. Sin embargo, el sensor de humedad del suelo HL-69 presentó inconvenientes con el paso del tiempo, esto causó que el riego de la motobomba no fuese preciso, por ello es recomendable utilizar un sensor de mayor calidad.
- Los informes estadísticos de las plantas ayudaron a visualizar los cambios en el cultivo de planta de fresa permitiendo un mayor control de sus condiciones ambientales ideales.
- La bitácora es útil para comprobar los cambios de estado de la planta provocado por lo actuadores a lo largo del tiempo y de esta manera estar al tanto de todas las acciones que suceden en el cultivo de fresa sin necesidad de estar presente en el lugar.
- Se realizó una prueba piloto utilizando los sensores: DHT11, HL-69, BH1750 para un cultivo pequeño de fresas en el que se realizaron cambios en las condiciones ambientales y actuó como se esperaba.
- Se desarrolló el sistema usando como referencia los diagramas UML diseñados para el software y los diagramas hechos en Fritzing y Proteus para el Hardware. Usando estos esquemas se logró la creación exitosa del prototipo de invernadero.
- Se logró hacer el despliegue de la página web en un servidor y conseguir un hosting y dominio público para poder acceder a la página desde cualquier dispositivo. El uso de Amazon Web Service permitió un almacenamiento de los componentes de la página de forma satisfactoria.

6. Recomendaciones

Como trabajo a futuro se podrían implementar las siguientes recomendaciones al prototipo presentado en este documento.

- Utilizar paneles solares con el fin de recargar la energía que alimenta el circuito y de esta manera ahorrar electricidad y evitar más conexiones en el proyecto.
- Implementar inicio de sesión con plataformas como Facebook, Google, entre otros.
- Validar que solo se puedan borrar/editar registros de crecimiento de plantas si estos registros fueron creados en el día de creado.
- Usar PCB para que el circuito quede más profesional y a su vez evitar la desconexión o ruptura de los cables UTP.
- Tener una tabla en la base de datos que registre las condiciones ideales de cada planta y de esta manera ampliar el rango de plantas que el proyecto puede controlar.
- Cambiar el sensor de humedad del suelo HL-69 por unos de mejor calidad dado que estos sensores presentaron problemas en las mediciones y proporcionaron valores con error.
- Desarrollar un sistema de notificaciones para la página web de acuerdo con las funciones de los actuadores.

- Implementar el uso de sensores de fuego para avisar al usuario que las plantas se encuentran en peligro o que controle un actuador que proporcione agua para apagar un posible incendio.
- Desarrollar una aplicación móvil que permita al usuario monitorear el estado de las plantas sin necesidad de acceder a la página. La aplicación puede tener funcionalidades adicionales como: el envío de mensajes de texto, correo, notificaciones de datos de interés para el usuario.
- Implementar en la página principal de la aplicación la visualización del último dato registrado en la base de datos para la planta en caso de tener el circuito apagado o desconectado de la red.
- Permitir una opción en la gestión de usuarios que permita agregar una foto de la persona en la página que podrá ver en el menú inferior izquierdo de la página principal.
- Cambiar la forma de recibir datos de la planta para que el usuario lo pueda decidir directamente desde la página y no tenga necesidad de cambiar el código de Arduino manualmente.
- Crear más circuitos para administrar una mayor cantidad de plantas y no tener la necesidad de mover el dispositivo.
- Mejorar la presentación del circuito para que se encuentre protegido del agua y tenga un diseño más compacto y atractivo.
- Utilizar inteligencia artificial para mejorar la toma de decisiones del sistema planteado, reduciendo la interacción humana en el invernadero.

Referencias Bibliográficas

- Agrimanager. (2019). *Tecnología y agricultura: por una producción sostenible en Colombia*. (Agrimanager) Obtenido de <https://cutt.ly/0SM7QdR>
- Agropinos. (06 de 03 de 2020). *¿CÓMO CULTIVAR FRESAS EN INVERNADERO?* Obtenido de <https://www.agropinos.com/blog/cultivo-de-fresas-en-invernadero>
- Agrositio. (2019). *¿Cómo se clasifican los cultivos?* Obtenido de <https://www.agrositio.com.ar/noticia/204187-como-se-clasifican-los-cultivo>
- Aguayo, P. (noviembre de 2004). *Introducción al Microcontrolador*. Obtenido de <https://cutt.ly/t83khhZ>
- Aharari, A., Yang, C., & Abe, J. (2021). Vegetable Distribution System Using Machine Learning Approach. In: Nakamatsu, K., Kountchev, R., Aharari, A., El-Bendary, N., Hu, B. (eds). *New Developments of IT, IoT and ICT Applied to Agriculture. Smart Innovation, Systems and Technologies, vol 183*. Springer, Singapore, 3-9. doi:https://doi.org/10.1007/978-981-15-5073-7_1
- Anónimo. (12 de 8 de 2021). Tendencias tecnológicas que transformarán la agricultura en Colombia. *Portafolio*. Obtenido de <https://cutt.ly/E87YLR2>
- Babiuch, M., Foltynek, P., & Smutny, P. (2019). Using the ESP32 Microcontroller for Data Processing. *International Carpathian Control Conference*. doi:<https://doi.org/10.1109/carpathiancc.2019.8765944>
- Calvo, A. (2019). *¿Qué es la agricultura tecnificada?* Obtenido de <https://www.agroptima.com/es/blog/agricultura-tecnificada/>
- Chaudhary, R., Nanda, J., & Tran, D. (2003). *Guía para identificar las limitaciones de campo en la producción de arroz*. Obtenido de <https://www.fao.org/3/y2778s/y2778s00.htm>
- Earth Observing System [EOS]. (2020). *Tipos De Cultivos Agrícolas: Maximizando Su Rendimiento*. Obtenido de <https://eos.com/es/blog/tipos-de-cultivos-agricolas/>
- Gridling, G., & Weiss, B. (2007). *Introduction To Microcontrollers*. Vienna University of Technology.
- Infoagro Systems, S.L. (s.f). *El cultivo de la Fresa*. Obtenido de https://www.infoagro.com/documentos/el_cultivo_fresa.asp#menuHeaderSectors
- Instituto Geográfico Agustín Codazzi [IGAC]. (2021). *Colombia, un país con una diversidad de suelos ignorada y desperdiciada*. Obtenido de <https://igac.gov.co/es/noticias/colombia-un-pais-con-una-diversidad-de-suelos-ignorada-y-desperdiciada>.
- Li, S., Xu, L., & Zhao, S. (2015). The internet of things: a survey. *Information Systems Frontiers*, 243–259. doi:<https://doi.org/10.1007/s10796-014-9492-7>

- Mantilla, J. (2013). *SISTEMAS Y SEÑALES*. Obtenido de http://sistemasyse.blogspot.com/2013/02/ejemplos-de-senales_6.html
- Martinez, S. (2014). *Metodología iterativa o incremental en la gestión de proyectos*. Obtenido de <https://www.mundoerp.com/blog/metodologia-iterativa-o-incremental-gestion-proyectos/>
- Pallás, A. R. (2003). *Sensores y acondicionadores de señal 4ª edición*.
- Pineda, J. (s.f.). *PRODUCTOS AGRICOLAS*. Obtenido de <https://encolombia.com/economia/agroindustria/agronomia/productos-agricolas/>
- Pineda, J. (s.f.). *TIPOS DE AGRICULTURA*. Obtenido de <https://encolombia.com/economia/agroindustria/tipos-de-agricultura/>
- Sabadí, P. (2017). *DESARROLLO ITERATIVO + MÍNIMO PRODUCTO VIABLE = SCRUM*. Obtenido de <https://cutt.ly/DS1ydxn>
- Singh, R. (2021). Recent Trends, Prospects, and Challenges of Nanobiosensors in Agriculture. In: Pudake, R.N., Jain, U., Kole, C. (eds). *iosensors in Agriculture: Recent Trends and Future Perspectives. Concepts and Strategies in Plant Sciences*. Springer, Cham. doi:https://doi.org/10.1007/978-3-030-66165-6_1
- Solectro. (2020). *Fundamentos de la señal analógica vs. la digital*. Obtenido de <https://cutt.ly/MDgcMPi>
- Stinson, L. (5 de 6 de 2014). A Smart Sensor That Quantifies the Soil in Your Garden. *WIRED*. Obtenido de <https://cutt.ly/N87TKdR>
- Thirugnanasambandan, T. (2021). Advances in Nanotechnology for Bio-Sensing in Agriculture and Food. In: Pudake, R.N., Jain, U., Kole, C. (eds). *Biosensors in Agriculture: Recent Trends and Future Perspectives. Concepts and Strategies in Plant Sciences*. Springer, Cham., 27-49. doi:https://doi.org/10.1007/978-3-030-66165-6_3
- Valdés, F., & Areny, R. (2007). *Microcontroladores fundamentos y aplicaciones con PIC (Vol. 1149)*. Marcombo.
- Vasey, D. (1992). *An Ecological History of Agriculture: 10,000 B.C. - A.D. 10,000*. Ames, Iowa: Iowa State: University Press.
- Zuppa, P. (2019). *CALIDAD Y TÉCNICAS DE EVALUACIÓN DE LOS SISTEMAS*. Obtenido de <https://cutt.ly/PS1w51P>

Apéndices

Apéndice A. Requerimientos del sistema.

A continuación, son presentadas las tablas con los requerimientos del sistema y sus respectivas especificaciones con el fin de esclarecer con mayores detalles la función que debe desempeñar cada requerimiento y la prioridad que se tuvo durante el desarrollo de este proyecto. Es importante aclarar que todos los requerimientos son necesarios para un correcto funcionamiento del sistema.

1. Requerimientos funcionales.

Tabla 6.

Registrar usuario

Requerimiento funcional	
Identificador (RF#)	RF01
Nombre	Registrar Usuario
Prioridad	Baja
Descripción	Los usuarios podrán crear usuarios indicando los nombres, apellidos, correo y una contraseña. Cabe resaltar que los usuarios no podrán ser eliminados.

Tabla 7.

Iniciar sesión

Requerimiento funcional	
Identificador (RF#)	RF02
Nombre	Iniciar sesión
Prioridad	Baja

Descripción	El usuario puede ingresar a la aplicación utilizando correo y la contraseña que haya especificado al momento de crear el usuario.
--------------------	---

Tabla 8.*Cerrar sesión*

Requerimiento funcional	
Identificador (RF#)	RF03
Nombre	Cerrar sesión
Prioridad	Baja
Descripción	Permitir cerrar sesión de la página web y redirigir a la página de log-in.

Tabla 9.*Editar usuario*

Requerimiento funcional	
Identificador (RF#)	RF04
Nombre	Editar usuario
Prioridad	Baja
Descripción	Permite cambiar los nombres, apellidos y contraseña del usuario. Para poder realizar esta acción la persona debe haber ingresado a la página con el usuario que desea realizar los cambios.

Tabla 10.*Agregar plantas*

Requerimiento funcional	
Identificador (RF#)	RF05
Nombre	Agregar plantas
Prioridad	Alta
Descripción	El usuario podrá crear plantas ingresando el nombre de la planta a crear. La planta se le asignará automáticamente un serial que servirá para enviar de datos desde el ESP32.

Tabla 11.*Modificar plantas*

Requerimiento funcional	
Identificador (RF#)	RF06
Nombre	Modificar plantas
Prioridad	Alta
Descripción	Permite cambiar el nombre de la planta registrada por uno diferente.

Tabla 12.*Visualizar estado de plantas*

Requerimiento funcional	
Identificador (RF#)	RF07
Nombre	Visualizar estado de plantas
Prioridad	Alta
Descripción	El usuario debe ser capaz de observar las plantas creadas, el estado de la planta y sus mediciones.

Tabla 13.*Medir datos de sensores*

Requerimiento funcional	
Identificador (RF#)	RF08
Nombre	Medir datos de sensores
Prioridad	Alta
Descripción	El hardware debe tomar los valores a partir de los sensores con su respectivo procesamiento para que lleven las unidades correctas.

Tabla 14.*Subir los datos de sensores a la nube*

Requerimiento funcional	
Identificador (RF#)	RF09
Nombre	Subir los datos de sensores a la nube

Prioridad	Alta
Descripción	El hardware debe enviar los datos a la nube a través del protocolo de comunicación MQTT para garantizar la seguridad y fiabilidad de los datos que sean recibidos del lado del servidor.

Tabla 15.*Activar y desactivar luces*

Requerimiento funcional	
Identificador (RF#)	RF10
Nombre	Activar y desactivar luces
Prioridad	Media
Descripción	El dispositivo debe cambiar el estado de las luces de acuerdo con los valores obtenidos por los sensores.

Tabla 16.*Activar y desactivar ventiladores*

Requerimiento funcional	
Identificador (RF#)	RF11
Nombre	Activar y desactivar ventiladores
Prioridad	Media
Descripción	El dispositivo debe cambiar el estado de los ventiladores de acuerdo con los valores obtenidos por los sensores.

Tabla 17.*Activar y desactivar riego*

Requerimiento funcional	
Identificador (RF#)	RF12
Nombre	Activar y desactivar riego
Prioridad	Media
Descripción	El dispositivo debe cambiar el estado de la motobomba de acuerdo con los valores obtenidos por los sensores.

Tabla 18.*Registrar crecimiento de la planta*

Requerimiento funcional	
Identificador (RF#)	RF13
Nombre	Registrar crecimiento de la planta
Prioridad	Media
Descripción	Permite al usuario registrar el valor del crecimiento registrado en la planta de su selección

Tabla 19.*Modificar crecimiento de la planta*

Requerimiento funcional	
Identificador (RF#)	RF14
Nombre	Modificar crecimiento de la planta
Prioridad	Media
Descripción	El usuario puede modificar el valor del crecimiento para la planta seleccionada por el usuario

Tabla 20.*Eliminar crecimiento de la planta*

Requerimiento funcional	
Identificador (RF#)	RF15
Nombre	Eliminar crecimiento de la planta
Prioridad	Media
Descripción	El usuario debe ser capaz de borrar el valor de crecimiento de la planta seleccionada.

Tabla 21.*Almacenar el histórico de acciones autónomas*

Requerimiento funcional	
Identificador (RF#)	RF16
Nombre	Almacenar el histórico de acciones autónomas
Prioridad	Media

Descripción	El hardware debe registrar las acciones autónomas realizadas en las plantas para poder ser enviadas al servidor.
--------------------	--

Tabla 22.*Generar informes de medición de sensores*

Requerimiento funcional	
Identificador (RF#)	RF17
Nombre	Generar informes de medición de sensores
Prioridad	Alta
Descripción	La aplicación debe presentar informes en forma de gráficos de las plantas de acuerdo con el sensor que realizó la medición. Las posibles gráficas son: temperatura, luminosidad, humedad del ambiente y humedad del suelo.

Tabla 23.*Generar informe de crecimiento de las plantas*

Requerimiento funcional	
Identificador (RF#)	RF18
Nombre	Generar informe de crecimiento de las plantas
Prioridad	Media
Descripción	La aplicación debe presentar informes en forma de gráficos de las plantas de acuerdo con el histórico de crecimiento registrado por el usuario.

Tabla 24.*Generar informe de histórico de acciones*

Requerimiento funcional	
Identificador (RF#)	RF19
Nombre	Generar bitácora de histórico de acciones
Prioridad	Media
Descripción	La aplicación debe presentar informes en forma de tablas de las plantas de acuerdo con el histórico de crecimiento registrado por el usuario.

2. Requerimientos no funcionales.

Tabla 25.*Permitir operaciones con varios usuarios al tiempo*

Requerimiento no funcional	
Identificador (RNF#)	RNF01
Nombre	Permitir operaciones con varios usuarios al tiempo
Prioridad	Media
Descripción	La página web debe permitir el ingreso de varios usuarios al mismo tiempo sin provocar efectos negativos en el aplicativo como errores con los datos y las vistas.

Tabla 26.*Diseñar interfaz gráfica de usuario*

Requerimiento no funcional	
Identificador (RNF#)	RNF02
Nombre	Diseñar interfaz gráfica de usuario
Prioridad	Alta
Descripción	Se debe diseñar una interfaz gráfica de usuario agradable e intuitiva que permita generar una buena experiencia a la hora de utilizar la aplicación.

Tabla 27.*Diseñar el circuito a utilizar*

Requerimiento no funcional	
Identificador (RNF#)	RNF03
Nombre	Diseñar el circuito a utilizar
Prioridad	Alta
Descripción	El esquema del circuito debe ser claro y coherente con las conexiones, sensores y elementos electrónicos para que a las personas puedan comprender y a su vez realizar una implementación del diseño planteado.

Tabla 28.*Realizar el montaje del circuito*

Requerimiento no funcional	
Identificador (RNF#)	RNF04
Nombre	Realizar el montaje del circuito
Prioridad	Alta
Descripción	El montaje del circuito es una de las partes fundamentales del proyecto, se trata del corazón de todas las operaciones con las plantas, encargada de realizar las mediciones e interactuar con los elementos que mantienen las condiciones ambientales ideales para un buen desarrollo del cultivo. Además de ello enviar datos al servidor

Tabla 29.*Conseguir hosting para el sistema*

Requerimiento no funcional	
Identificador (RNF#)	RNF05
Nombre	Conseguir hosting para el sistema
Prioridad	Alta
Descripción	El hosting servirá para almacenar la base de datos, almacenar el controlador de la página web tanto para las peticiones del Hardware como de la base de datos. En el proyecto se utilizó Amazon Web Service para llevar a cabo dicha función, como controlador se usó NodeJS y la base de datos fue MySQL. Adicional a ello, en Amazon también se encuentra el panel de control de Vesta y el panel de control de EMQX.

Tabla 30.*Conseguir dominio público para la aplicación*

Requerimiento no funcional	
Identificador (RNF#)	RNF06
Nombre	Conseguir dominio público para la aplicación
Prioridad	Alta
Descripción	El dominio público permitirá acceder a la página web desde cualquier dispositivo. Para el dominio público se recurrió a

Hostinger. Que permite usar certificados SSL y garantizar una conexión segura entre el usuario y el servidor.

Tabla 31.*Usar certificados SSL*

Requerimiento no funcional	
Identificador (RNF#)	RNF07
Nombre	Usar certificados SSL
Prioridad	Media
Descripción	El certificado SSL o seguridad en la capa de transporte en español hace que el dominio obtenga una comunicación segura por la red. Las páginas que poseen dicho certificado comienzan su enlace con “https”

Tabla 32.*Configurar servidor FTP*

Requerimiento no funcional	
Identificador (RNF#)	RNF08
Nombre	Configurar servidor FTP
Prioridad	Alta
Descripción	El protocolo FTP o protocolo de transferencia de archivos como su nombre lo indica es un servicio que permite el enviar y/o recibir archivos entre el servidor y un dispositivo, este protocolo es clave para el correcto funcionamiento de la página.

Tabla 33.*Utilizar el protocolo SSH*

Requerimiento no funcional	
Identificador (RNF#)	RNF09
Nombre	Utilizar el protocolo SSH
Prioridad	Alta
Descripción	El protocolo SSH o línea de comandos segura en español se encarga de brindar una conexión segura al servidor usando la red de internet. Esto evita que un tercero pueda acceder a la información mediante la encriptación de datos.

Tabla 34.*Hacer despliegue del aplicativo en un dominio público*

Requerimiento no funcional	
Identificador (RNF#)	RNF10
Nombre	Hacer despliegue del aplicativo en un dominio público
Prioridad	Media
Descripción	Tener la aplicación en un dominio público permitirá acceder a la página web gracias al DNS o sistema de nombres de dominios en español. Cuya función es redirigir a la IP deseada una vez se entra al dominio desde cualquier dispositivo, lugar y hora.

Tabla 35.*Implementar el protocolo MQTT*

Requerimiento no funcional	
Identificador (RNF#)	RNF11
Nombre	Implementar el protocolo MQTT
Prioridad	Alta
Descripción	El protocolo MQTT o cola de mensajes de transporte telemétricos en español se trata de un protocolo de comunicación entre maquinas, utiliza cifrado SSL para proteger los datos transmitidos entre maquinas. La implementación de este protocolo es parte fundamental del proyecto para que el servidor y el hardware se comuniquen de forma segura.

Tabla 36.*Conectar el hardware con la base de datos del servidor*

Requerimiento no funcional	
Identificador (RNF#)	RNF12
Nombre	Conectar el hardware con la base de datos del servidor
Prioridad	Alta
Descripción	Para realizar la conexión entre el servidor y el hardware se requiere que el módulo ESP32 se encuentre conectado a internet mediante Wi-Fi y tener el puerto, usuario, contraseña y la dirección del servidor MQTT.

Tabla 37.*Actualizar base de datos servidor*

Requerimiento no funcional	
Identificador (RNF#)	RNF13
Nombre	Actualizar base de datos servidor
Prioridad	Alta
Descripción	La base de datos del lado del servidor debe permitir la escritura y lectura de los datos que no sean solicitados por terceros. Es decir, permitir únicamente aquellas conexiones que hayan sido verificadas por los protocolos de seguridad implementados.

Tabla 38*Ser compatible con diferentes navegadores*

Requerimiento no funcional	
Identificador (RNF#)	RNF14
Nombre	Ser compatible con diferentes navegadores
Prioridad	Media
Descripción	La página web debe ser compatible principalmente con los siguientes navegadores: Google Chrome, Microsoft Edge, Opera, Firefox y otros navegadores basados en Chromium.

Apéndice B. Página web del proyecto desplegado.

A continuación, se presenta el enlace de la página web donde se encuentra el proyecto cargado en el servidor de Amazon Web Service. Cabe resaltar que otros usuarios no tienen permisos para cargar información desde un ESP32 a una planta de un usuario diferente. Es importante usar la cadena de conexión correcta a la hora de intentar subir información desde un ESP32 en el código de Arduino implementado. La cadena de conexión para enviar datos a la página web se encuentra en el repositorio GitHub.

<https://gardenstats.online/> o <http://3.218.111.159/>

Apéndice C. Repositorio GitHub.

El enlace GitHub del proyecto se encuentra a continuación, En la ruta se puede acceder a el código utilizado en Arduino y también el código fuente de la página web. El código de Arduino está ubicado dentro de la carpeta web, en una carpeta que se llama: Access_control. Es pertinente mencionar que en la raíz del directorio se encuentran las versiones anteriores del proyecto.

<https://github.com/HopeRiois/GardenStats/tree/main/gardenstats.ga>