



# UNIVERSIDAD NACIONAL DE LOJA

FACULTAD DE LA ENERGÍA, LAS  
INDUSTRIAS Y LOS RECURSOS NATURALES  
NO RENOVABLES

CARRERA DE INGENIERÍA EN  
ELECTRÓNICA Y TELECOMUNICACIONES

## “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA AUTOMATIZADO PARA PRODUCCIÓN DE PLÁNTULAS DE SIEMBRA EN EL CANTÓN LOJA”

TRABAJO DE TITULACIÓN ESPECIAL  
PREVIA A LA OBTENCIÓN DEL TÍTULO  
DE INGENIERO EN ELECTRÓNICA Y  
TELECOMUNICACIONES

**AUTOR:**

HUGO ADRIAN IDROVO CONTENTO

**TUTOR:**

ING. RODOLFO PABEL MERINO VIVANCO Mg. Sc.

LOJA - ECUADOR

2020

# CERTIFICACIÓN



**UNL**

Universidad  
Nacional  
de Loja

Facultad  
de la Energía, las Industrias y los  
Recursos Naturales No Renovables

## CARRERA DE INGENIERÍA EN ELECTRÓNICA Y TELECOMUNICACIONES

Certificado Nro. 2020-003-FEIRNNR-UNL

Ing. Rodolfo Pabel Merino Vivanco Mg.Sc.  
**DIRECTOR DE TESIS.**

### CERTIFICA:

Que el estudiante: **Hugo Adrian Idrovo Contento** con C.I: **1900597830**, aprobó el Trabajo de Titulación en el **11vo Ciclo** correspondiente al período académico: **07- Octubre-2019** al **06-Marzo-2020**; respecto del desarrollo de su tesis de grado titulada **"DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA AUTOMATIZADO PARA PRODUCCIÓN DE PLÁNTULAS DE SIEMBRA EN EL CANTÓN LOJA"**, opción de titulación escogida dentro del período académico de culminación de sus estudios, siendo las **14H00** del **04** de **febrero** del **2020**, se certifica que se ha cumplido con el cien por ciento (100%) del trabajo de titulación y está en condiciones de continuar con los procesos administrativos que correspondan.

Loja, 04 de febrero del 2020

Ing. Rodolfo Pabel Merino Vivanco Mg.Sc  
**DIRECTOR DE TESIS**

## **AUTORÍA**

Yo **HUGO ADRIAN IDROVO CONTENTO**, declaro ser autor del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la Publicaciones de mi trabajo de tesis en el Repositorio Institucional - Biblioteca Virtual

**Firma:**



**Cedula:** 1900597830

**Fecha:** 25/03/2020

## **CARTA DE AUTORIZACIÓN**

**CARTA DE AUTORIZACION TESIS POR PARTE DEL AUTOR, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO.**

Yo, **HUGO ADRIAN IDROVO CONTENITO** declaro ser autor de la tesis titulada **“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA AUTOMATIZADO PARA PRODUCCIÓN DE PLÁNTULAS DE SIEMBRA EN EL CANTÓN LOJA”**, como requisito para obtener el grado de: **INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional (RDI):

Los usuarios pueden consultar el contenido de este trabajo en RDI, en las redes de información del país y del exterior, con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la ciudad de Loja a los treinta días del mes de abril del dos mil veinte



**Firma:**

**Autor:** Hugo Adrian Idrovo Contento

**Cedula:** 1900597830

**Dirección:** Catacocha y 24 de Mayo

**Correo electrónico:** haidrovoc@unl.edu.ec - hugo.idrovo02@gmail.com

**Teléfono:** 072724448

**Celular:** 0985978892

### **DATOS COMPLEMENTARIOS**

**Director de Tesis:** Ing. Rodolfo Pabel Merino Vivanco Mg. Sc.

**Tribunal:** Ing. Manuel Augusto Pesantez González Mg. Sc.

Ing. Christian Hernán Campoverde Ramírez Mg. Sc.

Ing. Kleber Rolando Morillo Aguilar Mg. Sc.

## **DEDICATORIA**

El presente trabajo de titulación lo dedico con mucho cariño a Dios y a mi familia que siempre me ayudan, protegen, ellos son el motor que guía mi vida y están conmigo en todo momento.

Hugo Adrian Idrovo

## **AGRADECIMIENTO**

Agradezco a Dios por darme fortaleza y sabiduría para culminar esta tesis, a mi padre Hugo a mi madre Narcisa y a mi hermana Nathaly por su amor, su esfuerzo y apoyo incondicional.

A la Universidad Nacional de Loja a la Facultad de la Energía, las Industrias y los Recursos Naturales no Renovables y a todos los profesores de la Carrera de Ingeniería en Electrónica y Telecomunicaciones por toda la sabiduría impartida en las aulas de clase.

Al Ing. Pabel Merino, director del presente trabajo, por toda su guía y consejo para culminar con éxito esta tesis, al Ing. Manuel Pesantez por toda la ayuda brindada, así como también a los ingenieros Numan León y Vicente Amable por su asistencia en el taller mecánico.

A demás familia y amigos por su ayuda oportuna y desinteresada.

Hugo Adrian Idrovo

## TABLA DE CONTENIDO

PORTADA .....	I
CERTIFICACIÓN .....	II
AUTORÍA .....	III
CARTA DE AUTORIZACIÓN .....	IV
DEDICATORIA .....	V
AGRADECIMIENTO .....	VI
TABLA DE CONTENIDO .....	VII
ÍNDICE DE FIGURAS .....	XI
ÍNDICE DE TABLAS .....	XVI
<b>1. TÍTULO.....</b>	<b>1</b>
<b>2. RESUMEN .....</b>	<b>2</b>
<b>ABSTRACT.....</b>	<b>3</b>
<b>3. INTRODUCCIÓN .....</b>	<b>4</b>
<b>4. REVISIÓN DE LITERATURA .....</b>	<b>6</b>
<b>4.1. Variables físicas que controlan el crecimiento de plantas.....</b>	<b>6</b>
4.1.1. Factores que inciden en el crecimiento de la vegetación.....	6
4.1.1.1. Factores abióticos.....	6
4.1.1.2. Factores bióticos.....	7
4.1.2. Humedad.....	8
4.1.2.1. Importancia de la humedad del suelo en la calidad de los cultivos.....	8
4.1.2.2. La función de la humedad en el crecimiento de la planta .....	9
4.1.3. Ph.....	9
4.1.4. Temperatura.....	10
4.1.4.1. Influencia de la temperatura ambiental en las plantas.....	11
4.1.5. Viento .....	11
4.1.5.1. ¿Cómo se mide el viento? .....	12
4.1.5.2. Efecto del viento en las plantas .....	12

4.1.6.	Riego.....	13
4.1.7.	Presión atmosférica.....	14
4.1.7.1.	Los efectos de la presión atmosférica en las plantas.....	14
<b>4.2.</b>	<b>Hardware y software .....</b>	<b>15</b>
4.2.1.	Control numérico por computadora.....	15
4.2.2.	Plataforma de desarrollo electrónico .....	18
4.2.2.1.	Arduino.....	18
4.2.2.2.	<i>Shields</i> de arduino .....	19
4.2.3.	Computadora .....	19
4.2.3.1.	Arquitectura.....	19
4.2.3.2.	Clasificación de las computadoras .....	20
4.2.3.3.	Computadora de placa única (sbc) .....	20
4.2.4.	Sensor .....	23
4.2.4.1.	Tipos de sensores .....	23
4.2.5.	Actuador .....	24
4.2.5.1.	Tipos de actuadores.....	24
4.2.6.	Sistema operativo y software.....	24
4.2.6.1.	Sistema operativo .....	24
4.2.6.2.	Tipos de sistemas operativos.....	25
4.2.7.	Software.....	26
<b>4.3.</b>	<b>Interfaz web.....</b>	<b>26</b>
4.3.1.	Componentes .....	26
4.3.1.1.	Elementos de identificación: .....	26
4.3.1.2.	Elementos de navegación:.....	27
4.3.1.3.	Elementos de contenidos:.....	27
4.3.1.4.	Elementos de interacción: .....	27
4.3.2.	Factores de diseño .....	27
4.3.2.1.	Accesibilidad.....	27
4.3.2.2.	Funcionalidad .....	28
4.3.2.3.	Findability (encontrabilidad).....	28



4.3.2.4.	Utilidad.....	28
4.3.2.5.	Estética .....	28
4.3.2.6.	Credibilidad.....	29
4.3.2.7.	Usabilidad.....	29
<b>4.4.</b>	<b>Hosting y almacenamiento de información .....</b>	<b>29</b>
4.4.1.	<i>Hosting</i> o alojamiento web .....	29
4.4.1.1.	Tipos de hosting .....	30
4.4.1.2.	Host ddns.....	30
4.4.2.	Sql.....	31
4.4.2.1.	Base de datos.....	31
4.4.2.2.	Lenguaje del sql .....	31
4.4.2.3.	Sistemas de gestores de bases de datos (sgbd) comerciales y libres .	31
4.4.2.4.	Sistemas de gestores de bases de datos comerciales .....	32
4.4.2.5.	Sistemas de gestores de bases de datos libres .....	33
<b>5.</b>	<b>MATERIALES Y MÉTODOS.....</b>	<b>34</b>
<b>5.1.</b>	<b>Materiales .....</b>	<b>34</b>
5.1.1.	Materiales bibliotecarios y de oficina.....	34
5.1.1.1.	Materiales .....	34
<b>5.2.</b>	<b>Métodos.....</b>	<b>35</b>
5.2.1.	Especies vegetales del cantón loja.....	35
5.2.2.	Diseño del amazon del sistema robótico automatico .....	36
5.2.3.	Actuadores .....	43
5.2.4.	Equipos de monitoreo de variables físicas .....	49
5.2.4.1.	Dht22.....	49
5.2.4.2.	Y1-38.....	49
5.2.5.	Equipos de control .....	50
5.2.5.1.	Raspberry pi .....	50
5.2.5.2.	Arduino Mega .....	57
5.2.5.3.	Arduino Uno.....	59
5.2.5.4.	Arduino Nano .....	60

5.2.6.	Otros equipos .....	61
5.2.6.1.	Equipo de riego .....	61
5.2.7.	Software para control de actuadores y monitoreo de sensores .....	63
5.2.7.1.	Control de actuadores .....	63
5.2.7.2.	Software para monitoreo de sensores .....	65
5.2.7.3.	Software para automatización .....	67
5.2.7.4.	Lenguaje de programación para desarrollo web .....	76
5.2.7.5.	Software para servidor web y almacenamiento de datos .....	78
5.2.7.6.	Software DDNS .....	89
<b>6.</b>	<b>RESULTADOS .....</b>	<b>94</b>
<b>6.1.</b>	<b>Interfaz web .....</b>	<b>94</b>
6.1.1.	Usuario invitado .....	94
6.1.1.1.	Plantas .....	94
6.1.1.2.	Sensores .....	96
6.1.2.	Usuario administrador .....	97
<b>6.2.</b>	<b>Sistema automatizado .....</b>	<b>100</b>
6.2.1.	Siembra de semillas .....	101
6.2.2.	Riego .....	101
6.2.3.	Almacenamiento de información (base de datos) .....	102
6.2.3.1.	Sensor de temperatura/humedad ambiental .....	102
6.2.3.2.	Sensor de humedad de tierra .....	103
<b>7.</b>	<b>DISCUSIÓN .....</b>	<b>104</b>
<b>8.</b>	<b>CONCLUSIONES .....</b>	<b>107</b>
<b>9.</b>	<b>RECOMENDACIONES .....</b>	<b>108</b>
<b>10.</b>	<b>BIBLIOGRAFÍA .....</b>	<b>110</b>
<b>11.</b>	<b>ANEXOS .....</b>	<b>116</b>

## ÍNDICE DE FIGURAS

Figura 1. Factores de incidencia en el crecimiento de plántulas .....	7
Figura 2. (a) Anemómetro; (b) Manga de viento.....	12
Figura 3. Riego localizado en hidroponía.....	14
Figura 4. Diagrama de bloques de una maquina CNC .....	16
Figura 5. Funcionamiento de un sensor .....	23
Figura 6. Comunicación de SO con otras partes del sistema.....	25
Figura 7. Sistemas Operativos para: (a) computadoras y (b) dispositivos móviles.....	25
Figura 8. Relación entre programas de aplicación y programas del sistema.....	26
Figura 9. Factores de diseño orientado a la satisfacción-no frustración de uso .....	27
Figura 10. Una sala de servidores típica con cientos de máquinas.....	29
Figura 11. Funcionamiento del DDNS .....	30
Figura 12. Representación gráfica de una base de datos .....	31
Figura 13. Barra para base del armazón .....	37
Figura 14. a) Unión para las barras de la base; b) Barras unidas .....	37
Figura 15. Base del armazón terminado .....	37
Figura 16. a) Soporte para motor y barra calibrada; b) Soporte para perno y barra calibrada; c) Soporte de motor y barra ubicado en el armazón; d) Soporte de perno y barra ubicado en el armazón .....	38
Figura 17. Base del armazón con soportes de motores, pernos y barras calibradas .....	38
Figura 18. a) Soporte de ruedas y subiente; b) Barra subiente eje y; c) Soporte con ruedas ubicadas en base del armazón; d) Soporte con ruedas y barra de subida ubicado en armazón .....	39
Figura 19. Diseño del sistema robótico, en software de modelado 3D .....	39
Figura 20. (a) Base del armazón; (b) Platinas de unión.....	40
Figura 21. (a) Platinas para soporte de motores, pernos y barras calibradas; (b) Platinas de soporte <i>para ruedas</i> .....	40
Figura 22. (a) Ejes para las ruedas; (b) Torneado de los ejes de las ruedas .....	40
Figura 23. Platina, ejes y ruedas .....	41
Figura 24. (a) Soporte para motores; (b) Soporte para ruedas.....	41
Figura 25. Barra calibrada con anillo de seguridad .....	42
Figura 26. (a) Perno antes de ser torneado; (b) Perno luego de ser torneado .....	42

Figura 27. Armazón terminado.....	43
Figura 28. Motor pasa a paso Nema 23 de Pololu.....	46
Figura 29. Placa PCB creada en ARES del software Proteus .....	47
Figura 30. Placa PCB final .....	47
Figura 31. Motor dc .....	48
Figura 32. Motor Vibrador .....	48
Figura 33. Motor pasó a paso .....	48
Figura 34. Sensor de temperatura y humedad ambiental DHT22 .....	49
Figura 35. Sensor de humedad de tierra YL-38.....	49
Figura 36. Raspberry Pi 3 B+ .....	50
Figura 37. Sitio de descarga del SO Raspbian.....	51
Figura 38. Programa 7zip .....	51
Figura 39. Extrayendo archivo img de Raspbian .....	52
Figura 40. Página de BalenaEtcher.....	52
Figura 41. A) Acuerdo de licencia del programa BalenaEtcher; b) Instalación del programa BalenaEtcher .....	52
Figura 42. A) Pantalla principal del programa BalenaEtcher; b) Imagen de Raspbian seleccionada.....	53
Figura 43. A) MicroSD Samsung Evo Plus 64 GB; b) Ubicación de microSD en el programa BalenaEtcher .....	53
Figura 44. A) Instalando imagen de Raspbian en la microSD; b) Validando instalación .....	54
Figura 45. A) Instalación y validación finalizados; b) Ubicación de microSD en Raspberry Pi 3 B+.....	54
Figura 46. Ventana de configuración inicial de Raspbian.....	54
Figura 47. Configuración de país, idioma y zona de tiempo en configuración inicial de Raspbian .....	55
Figura 48. Establecimiento de nueva contraseña de Raspbian .....	55
Figura 49. Llenar pantalla en la configuración inicial de Raspbian .....	56
Figura 50. Elección de red WiFi en configuración inicial de Raspbian .....	56
Figura 51. Introducción de contraseña de red WiFi en configuración inicial de Raspbian .....	56

Figura 52. Actualización de software en configuración inicial de Raspbian .....	57
Figura 53. Reinicio de sistema en la configuración inicial de Raspbian.....	57
Figura 54. Placa de desarrollo Arduino Mega.....	58
Figura 55. Driver DRV8825 .....	58
Figura 56. Circuito impreso de la placa de soporte de driver.....	59
Figura 57. Placa de desarrollo Arduino Uno .....	60
Figura 58. Modulo driver ULN2003 .....	60
Figura 59. Placa de desarrollo Arduino Nano .....	61
Figura 60. Modulo driver L298N .....	61
Figura 61. Electroválvula .....	62
Figura 62. Sensor de nivel de agua EP2508 .....	62
Figura 63. Bomba de agua sumergible DC-1020 .....	63
Figura 64. Placa PCB para equipos de riego .....	63
Figura 65. Página en github de firmware GRBL para Arduino Mega .....	64
Figura 66. Programa en Arduino IDE para cargar el firmware GRBL .....	64
Figura 67. Programa para control de motores BYJ-24 en el entorno Arduino IDE .....	65
Figura 68. Programa de control de motores DC y vibradores en el entorno Arduino IDE .....	65
Figura 69. Página web GitHub de la librería Adafruit .....	66
Figura 70. Página web en GitHub de la librería DHT- sensor-libray.....	66
Figura 71. Programa para monitoreo de sensor DHT-22 .....	67
Figura 72. Ventana de instalación de CodeBlocks IDE, permiso de descarga y almacenamiento .....	68
Figura 73. Compilador gcc ya instalado en Raspbian .....	68
Figura 74. Comando de instalación de programas para descargar librería.....	69
Figura 75. Instalación de programas y librerías necesarias .....	69
Figura 76. Descarga de librería LibSerial.....	70
Figura 77. Configuración del compilador de CodeBlocks IDE.....	71
Figura 78. Ventana de búsqueda.....	71
Figura 79. Ventana "Linker settings" con los archivos agregados .....	72
Figura 80. Instalación de librería MySQL Connector C++ y librerías necesarias .....	72
Figura 81. Configuraciones globales del compilador de CodeBlocks.....	73

Figura 82. Librerías necesarias en CodeBlocks para MySQL Connector C++.....	73
Figura 83. Configuraciones globales del compilador con las librerías agregadas.....	74
Figura 84. Revisión de instalación de librería WiringPi .....	74
Figura 85. Ventana de configuración del compilador .....	75
Figura 86. Ventana de selección de archivos .....	75
Figura 87. Configuración del compilador con los archivos agregados .....	76
Figura 88. Ejemplo: Página de inicio de la Interfaz Web.....	77
Figura 89. Archivo bootstrap.css dentro del servidor web .....	77
Figura 90. Página web de descarga de Bootstrap .....	78
Figura 91. Confirmación de instalación Apache .....	79
Figura 92. Servicio de Apache encendido .....	79
Figura 93. Estado de firewall.....	80
Figura 94. Archivo de configuración del sitio web en Apache .....	81
Figura 95. Permiso de descarga e instalación Mariadb .....	82
Figura 96. Introducción de contraseña actual .....	82
Figura 97. Introducción de contraseñas nuevas.....	83
Figura 98. Eliminación de usuarios anónimos.....	83
Figura 99. Deshabilitar acceso remoto .....	84
Figura 100. Eliminación de bases de datos de prueba.....	84
Figura 101. Recargar privilegios de tablas .....	85
Figura 102. Configuración de PHPMyAdmin con respecto al servidor web .....	85
Figura 103. Tabla con usuario, cadena de autenticación, plugin y host en base de datos .....	86
Figura 104. Interfaz web de PHPMyAdmin.....	87
Figura 105. Interfaz para importar bases de datos PHPMyAdmin.....	88
Figura 106. Importación ejecutada exitosamente .....	88
Figura 107. Página web de No-IP.....	90
Figura 108. Creación de cuenta y dominio.....	90
Figura 109. a) Actualización de nombre de usuario; b) Actualización de pregunta de seguridad.....	91
Figura 110. Sección de descargas de la página web de no IP .....	91
Figura 111. Extraer el archivo descargado de la pagina web de No Ip .....	92

Figura 112. Configuración de programa de No-IP .....	92
Figura 113. Estado del programa de No-IP .....	93
Figura 114. Archivo para iniciar automáticamente No-IP.....	93
Figura 115. Interfaz web: Página de inicio usuario invitado .....	94
Figura 116. Interfaz web: Búsqueda de planta .....	95
Figura 117. Interfaz web: Selección de una planta.....	95
Figura 118. Interfaz web: Información de la planta .....	95
Figura 119. Interfaz web: Menú desplegable de Sensores .....	96
Figura 120. Interfaz web: Selección de intervalo de tiempo .....	96
Figura 121. Interfaz web: Gráfico de datos de sensor .....	97
Figura 122. Interfaz web: Cuadro de login.....	97
Figura 123. Interfaz web: Página de inicio usuario administrador.....	98
Figura 124. Interfaz web: Configuración inicial .....	98
Figura 125. Interfaz web: Configuración inicial selección de planta y dosificador .....	99
Figura 126. Interfaz web: Configuración inicial completa.....	99
Figura 127. Interfaz web: Control Manual .....	100
Figura 128. Sistema automatizado.....	101
Figura 129. Sistema de Siembra .....	101
Figura 130. a) Comprobación de humedad de tierra; b) Riego .....	102
Figura 131. Ejemplo de tabla Humedad Ambiental en base de datos .....	103
Figura 132. Ejemplo de tabla Humedad de tierra en base de datos.....	103

## ÍNDICE DE TABLAS

Tabla 1. Valores óptimos de desarrollo de hortalizas respecto al pH.....	10
Tabla 2. Códigos G y M para torno CNC.....	16
Tabla 3. Tipos de Arduino.....	18
Tabla 4. <i>Shields</i> de Arduino. ....	19
Tabla 5. Comparación de computadoras de placa única. ....	21
Tabla 6. Clasificación de los sensores .....	23
Tabla 7. Clasificación de los actuadores .....	24
Tabla 8. Materiales utilizados para el desarrollo del proyecto. ....	34
Tabla 9. Equipos utilizados para el desarrollo del proyecto.....	34
Tabla 10. Especies vegetales del cantón Loja .....	35
Tabla 11. Calculo de torque para elección de motor paso a paso.....	45



## **1. TÍTULO**

# **DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA AUTOMATIZADO PARA PRODUCCIÓN DE PLÁNTULAS DE SIEMBRA EN EL CANTÓN LOJA**

## 2. RESUMEN

El presente proyecto de tesis está orientado a la realización de un prototipo de sistema robótico automatizado que se encargará de la siembra y desarrollo de plántulas en una parcela de  $1\text{ m}^2$ , así como, la toma de datos mediante diferentes sensores que nos permitirán medir diferentes variables físicas como por ejemplo la temperatura y humedad ambiental o la humedad de la tierra, que pueden incidir en el crecimiento de la plántula, toda la información será almacenada en un *datalogger*, donde el usuario contará con el diseño de una interfaz web de fácil acceso y manejo que le ayudará en la gestión del sistema remotamente.

El proyecto se inicia con la búsqueda del estado del arte tanto de los parámetros que necesita una planta para crecer de manera óptima, como de los diferentes softwares y hardwares que se usaron; el software y hardware utilizado en el proyecto de tesis son de libre difusión (Open-Source) que vienen en una variedad de equipos y programas. Esta implementación se la puede hacer escalable, dependiendo de los recursos económicos y de financiamiento, el proyecto está caracterizado para investigar en pequeñas parcelas para la siembra y germinación de plántulas, hasta espacios más grandes como invernaderos donde la comercialización de plántulas sería a gran escala.

Al final el sistema automatizado facilitará la tarea de la generación de plántulas ya que el sistema se encargará de sembrar, regar y monitorear las variables de germinación de la plántula, ofreciendo al usuario ahorro de tiempo, mejor control del uso del agua y suelo, a grandes escalas ahorro en contratación de personal para cultivo y a la larga un retorno del capital invertido en el sistema.

## **ABSTRACT**

The present thesis project is oriented at the realization of a prototype automated robotic system that will be responsible for the sowing and developing seedlings on a plot of 1 m<sup>2</sup>, as well as, the take of data through different sensors that will allow us measure different physical variables such as environmental temperature and humidity or the humidity of the earth, which can influence the growth of the seedling, all the information will be stored in a datalogger, where the user will count with the design of an easily accessible web interface and handling that will help it in the system management remotely.

The project begins with the search for the state of the art of the parameters that a plant needs to grow optimally, as well as the different software and hardware used; the software and hardware used in the thesis project are of free diffusion (Open Source) that come in a variety of equipment and programs. This implementation can be made scalable, depending of economic resources and financing, the project is characterized to investigate in small plots for sowing and germination of seedlings, until larger spaces such as greenhouses where the commercialization would be large-scale.

In the end, the automated system will facilitate the task of seedling generation since the system will be in charge of sowing, to water and monitoring the germination variables of the seedling, offering the user save time, better control of water and soil use, to large-scales, savings in hiring personnel for cultivation and in the long run a return on capital invested in the system.

### 3. INTRODUCCIÓN

Desde la antigüedad las personas han cultivado por la sencilla razón de alimentarse, era una tarea artesanal, pero en los tiempos que corren esta actividad se desarrolla a nivel global las personas han hecho de esto una actividad productiva generadora de divisas en muchos países, en la actualidad. Pero debido a un mundo cada vez mayor esta actividad como se la ha venido realizando hace tantos años se ha ido complicando por lo que se han visto en la necesidad del uso de nuevas tecnologías, para poder alimentar a cada vez más personas ocupando cada vez menos recursos (terrenos, insumos agrícolas, agua, etc.). (Nates, 2013)

También la agricultura por años ha hecho un mal uso de sus recursos, la tecnificación de la agricultura no ha propiciado un mejor uso de los terrenos, el uso indiscriminado de pesticidas nocivos para la salud y el desperdicio del agua que en el tiempo termina erosionando el terreno y a la vez pérdidas económicas a los propietarios de los terrenos.

Existe un término que cada vez va a ir ganando más importancia la agrotecnología, es decir la combinación de la agricultura con la tecnología que cada vez toma más importancia en nuestras vidas y la agricultura no es una excepción. La agrotecnología no es un término reciente en su significado, la agrotecnología según Hermosilla “Ha evolucionado pasando por varias fases o etapas, que van desde las civilizaciones más antiguas —arados tirados por bestias— hasta los robots y drones empleados hoy en día en el mejoramiento de la producción agrícola, con un menor impacto medioambiental y social.” (Hermosilla Galeano, 2015)

Debido a la tecnología se han realizado grandes y muy variados avances y muchos de se han creado o adaptado para según Hermosilla “facilitar la agricultura con el uso de la tecnología, tenemos por ejemplo: sembradoras y tractores con GPS, Software para la agricultura de alta precisión, Tecnología en la recolección, Drones y robots para la agricultura” (Hermosilla Galeano, 2015)

Al implementar el sistema se logrará un mejor control en las variables que afectan la germinación y crecimiento de las semillas hasta la obtención de las plántulas dado que realizará el cuidado necesario.

Por lo que el presente proyecto de tesis pretende hacer uso de uso de la agrotecnología dado que se diseñará e implementará en una parcela de 1 m x 1 m un sistema automatizado

para siembra, riego y monitoreo de variables (temperatura, humedad ambiental, humedad de tierra) para producir plántulas que luego pueden ser trasplantadas.

El presente trabajo de tesis se orientó a cumplir el siguiente Objetivo General: “Diseñar e implementar un sistema automatizado en una parcela para siembra de semillas y crecimiento de plántulas, utilizando dispositivos electrónicos Open-Source para controlar actuadores y monitorear sensores mediante la creación de una interfaz web”. Mientras que se definió los Objetivos Específicos como:

1. Determinar los parámetros para el crecimiento óptimo de plántulas de siembra en el cantón Loja.
2. Determinar y describir las características técnicas del sistema, equipos y software libre necesario para el correcto funcionamiento del sistema robótico.
3. Diseñar y construir el sistema robótico automatizado incorporando hardware con asistencia de software CNC Open-Source.
4. Diseñar la interfaz web para el monitoreo y control del sistema utilizando herramientas de desarrollo web y albergar la interfaz web en un servidor conectado a la red.

## **4. REVISIÓN DE LITERATURA**

### **4.1. Variables físicas que controlan el crecimiento de plantas**

#### **4.1.1. Factores que inciden en el crecimiento de la vegetación**

Según Reyes “La productividad agrícola para el desarrollo y crecimiento de las plantas depende de algunos factores” (Reyes Mejia, 2013).

##### **4.1.1.1. Factores abióticos**

“Entre los factores físicos” están (El medio ambiente, s.f.):

- Elementos del clima; la luz, la temperatura, el agua y el viento:
  - La luz que utilizan las plantas para realizar la fotosíntesis (El medio ambiente, s.f.).
  - La temperatura un elemento que permite que la planta realice sus diferentes procesos como son crecimiento y floración (El medio ambiente, s.f.).
  - El agua dadora de vida, en las plantas la utilizan para junto con ella absorber los nutrientes de la tierra (El medio ambiente, s.f.).
  - Los vientos que dan inicio a diferentes procesos como son llevar las semillas controlar la humedad y temperatura (El medio ambiente, s.f.).
- La composición del suelo:
  - La composición que está formada variada materia orgánica e inorgánica que la pueden hacer infértil o no, esta composición permite que la planta pueda crecer (El medio ambiente, s.f.).
- La geografía (El medio ambiente, s.f.):
  - La altitud tiene influencia en el crecimiento de las plantas y los diferentes elementos antes citados (El medio ambiente, s.f.).
  - La latitud o ubicación geométrica dado que ello determina los tiempos de las diferentes estaciones del año (El medio ambiente, s.f.).
  - La orientación, en una montaña le puede generar a la planta diferentes condiciones de sol y lluvia (El medio ambiente, s.f.).
  - “La mayor inclinación de una ladera aumenta la erosión por arrastre de la lluvia y vientos” (El medio ambiente, s.f.).

#### 4.1.1.2. Factores bióticos

Es decir como las plantas, los animales y microorganismos se relacionan entre ellos para poder subsistir (El medio ambiente, s.f.).

En las plantas, intervienen (El medio ambiente, s.f.):

- Los microorganismos que descomponen los nutrientes del suelo (El medio ambiente, s.f.).
- Otras plantas que pueden ser beneficiosas o desfavorables entre ellas (El medio ambiente, s.f.).
- Los animales que se alimentan de ellas y luego dejan sus desechos, algunos también ayudan a que las plantas generen sus frutos o lleven sus semillas a otros lugares (El medio ambiente, s.f.).

En los animales influye (El medio ambiente, s.f.):

- “La disponibilidad de alimento” (El medio ambiente, s.f.).
- Los animales entre si (El medio ambiente, s.f.).

“En los humanos o aspectos sociales” (El medio ambiente, s.f.):

Los seres humanos como forma dominante y para poder vivir se aprovechan de todas las demás formas de vida o las ayudan en beneficio mutuo (El medio ambiente, s.f.)

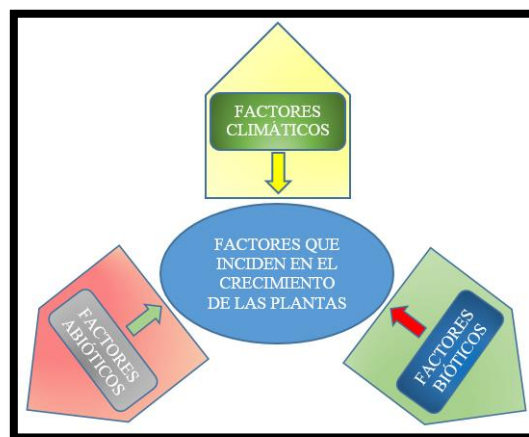


Figura 1. Factores de incidencia en el crecimiento de plántulas

Fuente: [El Autor]

#### 4.1.2. Humedad

“La humedad es un factor climatológico que se define como vapor de agua contenido en la atmósfera. Este vapor de agua permite la formación de las nubes, las cuales a su vez colaboran con la humedad del ambiente, cuando al condensarse se precipitan a la tierra en forma de lluvia o nieve” (Definición de Humedad, s.f.).

La humedad al ser un elemento físico se lo puede medir y para ello se vale de un instrumento denominado psicrómetro que son dos termómetros uno usado de manera normal y el otro con una cubierta mojada puesto en un contenedor de agua. (Rodríguez Jiménez, Capa, & Portela Lozano, 2004).

Existen diversas maneras de referirnos al contenido de humedad en la atmósfera:

- **Humedad Relativa:** “es la relación que existe entre la cantidad de vapor existente en el aire y la que debería tener para saturarse a igual temperatura” (Definición de Humedad, s.f.).
- **Humedad Absoluta:** “masa de vapor de agua, en gramos, contenida en 1m de aire seco” (Rodríguez Jiménez, Capa, & Portela Lozano, 2004).
- **Humedad Específica:** “masa de vapor de agua, en gramos, contenida en 1 kg de aire” (Rodríguez Jiménez, Capa, & Portela Lozano, 2004).
- **Razón de Mezcla:** “masa de vapor de agua, en gramos, que hay en 1 kg de aire seco” (Rodríguez Jiménez, Capa, & Portela Lozano, 2004).

##### 4.1.2.1. Importancia de la humedad del suelo en la calidad de los cultivos

La humedad en lo que respecta al suelo se refiere según González “a la cantidad de agua cerca de la superficie del suelo y corresponde al agua contenida en los primeros centímetros.” (González Robles, 2012).

La humedad varia dado que en ninguna parte las condiciones son iguales por lo que la influencia de diferentes factores hace que el suelo cambie, el tipo de vegetación la altura entre otras. (Alvarez-Mozos, Casalí, González-Audicana, & López, 2005).

La humedad en los invernaderos según Peery “puede ser el factor ambiental más difícil de controlar”. Debido a la magnitud de estos, aunque se utilicen equipos especiales, todavía existe un gran margen de error en el control de variables, dado a que las plantas



influyen también en el ambiente transpiran y agregan vapor de agua al ambiente constantemente (Peery, 2017).

#### **4.1.2.2. La función de la humedad en el crecimiento de la planta**

Uno de los factores que intervienen en el desarrollo de las plantas es por supuesto la humedad, debido a que si esta se encuentra en los extremos tanto superior como inferior hará que la planta sufra en todos los aspectos, alimentación, crecimiento, etc. (Peery, 2017).

La humedad del aire afecta a las plantas e interviene en según Peery “la transpiración y la fotosíntesis” (Peery, 2017):

- **Transpiración:** “Es el proceso donde las plantas absorben agua a través de las raíces y, luego, producen vapor de agua a través de los poros de sus hojas.” (Peery, 2017)
- **Fotosíntesis:** “Es el proceso de fijar el dióxido de carbono y el agua en las hojas de las plantas para producir azúcares que se usan para la energía y el crecimiento. La calidad del cultivo depende de las condiciones que promuevan una fotosíntesis óptima, y la humedad desempeña una función en este proceso” (Peery, 2017).

#### **4.1.3. Ph**

El pH en latín cantidad de hidrogeno es una medida que nos permite saber si una solución es acida u alcalina, esta medida se encuentra entre un mínimo 0 que significa que es muy acida hasta ubicarse en un valor cercano a 7, un valor de 7 indica que es un solución neutra, mientras que un valor mayor a 7 y máximo 14 significa una solución alcalina (La importancia de la acidez del pH para tus plantas, s.f.).

Para que una planta se desarrolle correctamente necesita que la tierra en la que está sembrada se encuentre a un nivel óptimo de pH (varía según cada planta) este nivel se encuentra en la parte acida esto es porque la acides en la tierra aporta solubilidad de los nutrientes y de la existencia de diferentes microorganismos beneficiosos para las plantas (La importancia de la acidez del pH para tus plantas, s.f.).

“El control y regulación del pH del suelo es importante. En la Tabla 1 mostramos algunos valores óptimos de desarrollo de hortalizas con respecto al pH. El verde más oscuro refleja el pH de las plantas óptimo, mientras que el verde más apagado muestra el desarrollo

aceptable o aquel que no suponga ningún problema de crecimiento para la planta” (El pH de las plantas, s.f.).

Tabla 1. Valores óptimos de desarrollo de hortalizas respecto al pH

Cultivo	pH							
	5	5,5	6	6,5	7	7,5	8	8,5
Alcachofa								
Apio								
Berenjena								
Calabaza								
Cebolla								
Col								
Coliflor								
Espinaca								
Guisante								
Habas								
Judía								
Lechuga								
Maíz								
Melón								
Patata								
Pepino								
Pimiento								
Rábano								
Tomate								
Zanahoria								

Fuente: (El pH de las plantas, s.f.)

#### 4.1.4. Temperatura

Según Rodríguez, Capa & Portela “La Temperatura es una de las magnitudes más utilizadas que describe el estado de la atmosfera. Sabemos que la temperatura del aire varía entre el día y la noche, entre una estación y otra, y también entre una ubicación geográfica y otra” (Rodríguez Jiménez, Capa, & Portela Lozano, 2004).

La temperatura se produce debido a la interacción de los átomos y moléculas que existen en la materia dado a que estas al nivel de interacción de su movimiento un objeto tangible o intangible generara calor o el contrario (Rodríguez Jiménez, Capa, & Portela Lozano, 2004).

Como todo elemento físico este se puede medir y aprovechando que la materia se ve influenciada por los elementos y actúa de diferente manera según Rodríguez, Capa & Portela “la resistencia eléctrica de algunos materiales, el volumen de un cuerpo, el color

de un objeto, etc. El instrumento que se utiliza para medir la temperatura es el termómetro” (Rodríguez Jiménez, Capa, & Portela Lozano, 2004). “Actualmente se utilizan tres escalas de temperatura; grados Fahrenheit (°F), Celsius (°C) y Kelvin (°K)” (Definición de Temperatura, s.f.).

La temperatura se ve influenciada por la altitud ya que según donde se mida más altura la temperatura tiende a bajar y a menor altura tiende a subir, también influye el lugar es decir las condiciones topológicas y zonas geográficas, esta temperatura permite o no que se desarrolle la vida (Definición de Temperatura, s.f.).

#### **4.1.4.1. Influencia de la temperatura ambiental en las plantas**

“Las temperaturas tienen efecto sobre la velocidad de crecimiento, germinación, transpiración, respiración, fotosíntesis, y absorción de agua y nutrientes” (Acción de la temperatura sobre la vegetación, s.f.).

En las plantas el aumento de la temperatura repercute para bien o para mal dependiendo desde la perspectiva que se vea dado que la planta comenzará a desarrollarse más rápidamente, pero también al suceder esto la planta verá su energía agotarse y esto solo logrará que la planta no de buenos frutos a la larga (Influencia de la temperatura ambiental en las plantas, s.f.).

“Las plantas buscan alcanzar su temperatura óptima, para lo que es muy importante que exista un equilibrio entre la temperatura ambiental, la humedad relativa y la luz. Si los niveles de luz son altos la planta se calentará demasiado, produciéndose una diferencia entre la temperatura ambiental y la de la planta. Para enfriarse, el índice de transpiración de la planta deberá aumentar” (Influencia de la temperatura ambiental en las plantas, s.f.).

Debido a que la temperatura varía en función de la hora la planta se vale de esto para en las temperaturas más bajas que se producen en la noche enviar nutrientes a las partes de la planta como son flores y frutos para que se desarrollen mejor (Influencia de la temperatura ambiental en las plantas, s.f.).

#### **4.1.5. Viento**

Según Rodríguez, Capa & Portela “El viento consiste en el movimiento de aire desde una zona hasta otra. Existen diversas causas que pueden provocar la existencia del viento,

pero normalmente se origina cuando entre dos puntos se establece una cierta diferencia de presión o de temperatura” (Rodríguez Jiménez, Capa, & Portela Lozano, 2004).

#### 4.1.5.1. ¿Cómo se mide el viento?

El viento es medible de dos maneras diferentes dado a que se lo puede realizar para medir tanto su intensidad medida como velocidad y su dirección sea esta según el punto cardinal del que provenga, según Rodríguez, Capa & Portela “Para medir la velocidad horizontal del viento, el instrumento más utilizado es el anemómetro” (Figura 2 (a)) “en el que el giro de las mismas es proporcional a la velocidad del viento. La unidad de medida es el km/h o el m/s. Para medir la dirección, por otro lado se utilizan las veletas o mangas” (Figura 2 (b)), “que indican la procedencia geográfica del viento. Hablamos de viento norte, noreste, suroeste, etc., en función de donde provenga este” (Rodríguez Jiménez, Capa, & Portela Lozano, 2004).



(a)



(b)

Figura 2. (a) Anemómetro; (b) Manga de viento

Fuente: (Qué es el viento, cómo se produce y los diferentes tipos, s.f.)

#### 4.1.5.2. Efecto del viento en las plantas

Debido al viento este puede hacer que la planta se seque, a parte la afecta también indirectamente ya que afecta el suelo en el que está sembrada porque tiene el mismo efecto sobre este; según la intensidad del mismo también puede dañar irremediablemente a la planta (¿Cómo afecta el viento en las plantas?, 2016).

“El viento como acción positiva permite el transporte de polen y la diseminación de semillas, pero además tiene un papel importante en los tejidos de los troncos y ramas, cuando hay viento suave, las plantas se mueven y los tejidos se rompen, creando una

respuesta de tejidos más fuertes, con el consiguiente plantas cada vez más fuertes y firmes, como acción negativa fuertes vientos pueden aumentar el efecto de la limitación hídrica, el efecto mecánico puede llegar provocar la abscisión de hojas y de frutos” (¿Cómo afecta el viento en las plantas?, 2016).

En otros casos, las plantas han detenido su altura o bien adquirido formas más redondeadas para no sufrir los estragos de las ráfagas que atentan contra su integridad. Estas alternativas colaboran además a la hora de evitar la deshidratación (¿Cómo afecta el viento en las plantas?, 2016).

#### **4.1.6. Riego**

Uno de los factores más importantes también es el riego ya que debido a este la planta se hidrata y puede absorber de mejor manera los alimentos por las raíces. “Ésta es una actividad necesaria tanto en la hidroponía, como en la agricultura tradicional y la jardinería” (¿Qué es el riego?, s.f.).

Se puede hacer tanto por la acción de la naturaleza es decir la lluvia o dependiendo de la región de no ser posible esto entonces se acude a el uso de otros medios que la gente ha ido desarrollando a lo largo de los siglos para dar sustento hídrico a sus cosechas (¿Qué es el riego?, s.f.).

“En el caso de la hidroponía el riego no solo aporta agua si no también los nutrientes, que le son proporcionados a la planta a través de la solución nutritiva logrando su desarrollo integral mediante el riego que le sea suministrado” (¿Qué es el riego?, s.f.).

Las técnicas de riego más utilizadas son:

- “Arroyamiento o surcos.”
- “Inundación.”
- “Riego localizado.”
- “Riego por aspersión.”
- “Riego por drenaje” (¿Qué es el riego?, s.f.)

No describiremos todas en este trabajo, pero si, enfocaremos una variable a controlar, la humedad del suelo para lo cual el riego localizado es el adaptado a nuestro trabajo

Este riego puede ser de tres maneras, pero todas cumplen con las mismas funciones (¿Qué es el riego?, s.f.):

- “No moja todo el suelo.”
- “Son pequeñas dosis de agua, que se aplican muchas veces.”
- “Mantienen el suelo siempre húmedo.”
- “Va orientado a satisfacer las necesidades de la planta y no del suelo” (¿Qué es el riego?, s.f.)



Figura 3. Riego localizado en hidroponía

Fuente: (¿Qué es el riego?, s.f.)

#### **4.1.7. Presión atmosférica**

Según Rodríguez, Capa & Porte “el aire que nos rodea, pesa y, por tanto, ejerce una fuerza sobre todos los cuerpos debida a la acción de la gravedad. Esta fuerza por unidad de superficie es la denominada presión atmosférica, cuya unidad de medida en el Sistema Internacional es el Pascal (1 Pascal =  $1\text{N/m}^2$ )” (Rodríguez Jiménez, Capa, & Portela Lozano, 2004)

Como todo factor físico la presión atmosférica es factible de ser medida, y el instrumento creado para este fin es el llamado barómetro.

##### **4.1.7.1. Los efectos de la presión atmosférica en las plantas**

Según Turtenwald “la presión atmosférica en la Tierra es de aproximadamente 101 kilopascales, o kPa, que es la condición ideal para que las plantas puedan crecer. Una de las razones por las que las plantas tienen problemas para crecer en otros planetas es la diferencia en la presión atmosférica” (Turtenwald, 2018).

- **Velocidad de crecimiento**

La presión atmosférica cambia según la altura a la que se encuentre, esto hace que la planta se desarrolle de manera diferente y también a la que la planta esté o no adaptada a dicha presión, aunque esto no evita que algunas planta crezcan si define la velocidad a la que lo hacen o definitivamente nunca prospere debido a la falta concentración de oxígeno y dióxido de carbono en el aire (Turtenwald, 2018).

- **Concentración de gas**

Según Tunderwald “la concentración de gas es más importante que el valor de la presión atmosférica. En la atmósfera terrestre el oxígeno constituye alrededor del 20% de los gases presentes en el aire, necesario para el crecimiento de las plantas. Incluso si reduces o aumentas la presión atmosférica es importante mantener la misma cantidad de oxígeno para su crecimiento, que es aproximadamente de 15 a 20 kPa sin importar la cantidad de presión” (Turtenwald, 2018).

## **4.2. Hardware y software**

### **4.2.1. Control numérico por computadora**

Se trata de un sistema permite mover un objeto y mantener control de posición sobre él para cumplir un propósito, ya que esto suele estar ligado a distintos tipos de herramientas como son fresadoras, extrusoras entre otras, esto se logra usando un software dedicado para ello y un computadora que ponga en funcionamiento el software y envíe código a la máquina (¿Qué significa cnc?, s.f.).

“El control numérico computarizado es el uso de una computadora para controlar y monitorear los movimientos de una máquina herramienta ... El controlador CNC trabaja en conjunto con una serie de motores (servomotores y/o motores paso a paso), así como componentes de accionamiento para desplazar los ejes de la máquina de manera controlada y ejecutar los movimientos programados” (Introducción a la tecnología CNC, 2015).

En la Figura 4. Diagrama de bloques de una maquina CNC podemos observar las distintas partes de las que puede estar formado una maquina CNC

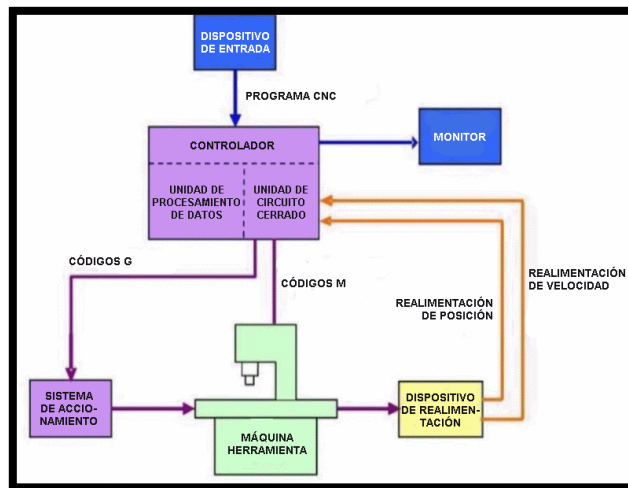


Figura 4. Diagrama de bloques de una máquina CNC

Fuente: (Introducción a la tecnología CNC, 2015)

En la figura se muestra mediante diagrama de bloques como es que se desarrolla la comunicación entre los diferentes elementos que conforman el sistema, estos van desde introducir conjuntos de comandos a la computadora, esta se encargara de transformar esos comandos en códigos G o M para que la máquina CNC haga su trabajo (Introducción a la tecnología CNC, 2015).

El código G se lo usa para que la máquina se mueva, es decir cambie de posición con respecto a un sistema que puede ser el cartesiano, y el código M en su lugar se refiere al uso de herramienta en la máquina como puede ser un láser, agua, broca, etc. (Introducción a la tecnología CNC, 2015).

En la Tabla 2 podemos observar algunos ejemplos de los códigos G y M utilizados en una máquina CNC para un torno

Tabla 2. Códigos G y M para torno CNC.

CÓDIGOS G	CÓDIGOS M
G00: Posicionamiento rápido (sin maquinar)	M00: Parada opcional
G01: Interpolación lineal (maquinando)	M01: Parada opcional
G02: Interpolación circular (horaria)	M02: Reinicio del programa
G03: Interpolación circular (antihoraria)	M03: Hacer girar el husillo en sentido horario
G04: Compás de espera	M04: Hacer girar el husillo en sentido antihorario
G10: Ajuste del valor de offset del programa	M05: Frenar el husillo
G20: Comienzo de uso de unidades imperiales (pulgadas)	M06: Cambiar de herramienta
	M07: Abrir el paso del refrigerante B
	M08: Abrir el paso del refrigerante A
	M09: Cerrar el paso de los refrigerantes
	M10: Abrir mordazas



G21: Comienzo de uso de unidades métricas	M11: Cerrar mordazas
G28: Volver al home de la máquina	M13: Hacer girar el husillo en sentido horario y abrir el paso del refrigerante
G32: Maquinar una rosca en una pasada	M14: Hacer girar el husillo en sentido antihorario y abrir el paso del refrigerante
G36: Compensación automática de herramienta en X	M30: Finalizar programa y poner el puntero de ejecución en su inicio
G37: Compensación automática de herramienta en Z	M31: Incrementar el contador de partes
G40: Cancelar compensación del radio de curvatura de herramienta	M37: Frenar el husillo y abrir la guarda
G41: Compensación del radio de curvatura de herramienta a la izquierda	M38: Abrir la guarda
G42: Compensación del radio de curvatura de herramienta a la derecha	M39: Cerrar la guarda
G70: Ciclo de acabado	M40: Extender el alimentador de piezas
G71: Ciclo de maquinado en torneado	M41: Retraer el alimentador de piezas
G72: Ciclo de maquinado en frentado	M43: Avisar a la cinta transportadora que avance
G73: Repetición de patrón	M44: Avisar a la cinta transportadora que retroceda
G74: Taladro intermitente, con salida para retirar virutas	M45: Avisar a la cinta transportadora que frene
G76: Maquinar una rosca en múltiples pasadas	M48: Inhabilitar Spindle y Feed override (maquinar exclusivamente con las velocidades programadas)
G96: Comienzo de desbaste a velocidad tangencial constante	M49: Cancelar M48
G97: Fin de desbaste a velocidad tangencial constante	M62: Activar salida auxiliar 1
G98: Velocidad de alimentación (unidades/min)	M63: Activar salida auxiliar 2
G99: Velocidad de alimentación (unidades/revolución)	M64: Desactivar salida auxiliar 1
	M65: Desactivar salida auxiliar 2
	M66: Esperar hasta que la entrada 1 esté en ON
	M67: Esperar hasta que la entrada 2 esté en ON
	M70: Activar espejo en X
	M76: Esperar hasta que la entrada 1 esté en OFF
	M77: Esperar hasta que la entrada 2 esté en OFF
	M80: Desactivar el espejo en X
	M98: Llamada a subprograma
	M99: Retorno de subprograma
Fuente: (Introducción a la tecnología CNC, 2015)	

Pudiendo estos códigos ser combinados en bloques para que posteriormente la máquina CNC realice los distintos procedimientos conforme a los códigos G y M por ejemplo:

**M02 G01 X25.000 Z32.000**

Siendo:

- M02: Reinicio del programa
- G01: Interpolación lineal, hacia las coordenadas X=25 mm y Z=32 mm.

## 4.2.2. Plataforma de desarrollo electrónico

### 4.2.2.1. Arduino

“Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios” (Qué es Arduino, s.f.).

Arduino es usado para muchos propósitos dado a que es un sistema muy versátil da a que se puede desarrollar múltiples proyectos electrónicos con él, desde crear pianos y robots, hasta usar sensores o apagar y encender un foco.

Según Doutel “Podemos catalogar los usos que podemos darle a Arduino en dos grandes grupos”:

- “Aquellos en los que el Arduino es utilizado como microcontrolador, tiene un programa descargado desde un ordenador y funciona de forma independiente de éste, y controla y alimenta determinados dispositivos y toma decisiones de acuerdo al programa descargado e interactúa con el mundo físico gracias a sensores y actuadores” (Doutel, 2017).
- “La placa Arduino hace de interfaz entre un ordenador (como podría ser una Raspberry Pi) u otro dispositivo, que ejecuta una determinada tarea, para traducir dicha tarea en el mundo físico a una acción (actuadores). Y viceversa, gracias a sensores que están conectados a la placa Arduino podemos hacer que el ordenador ejecute determinada acción” (Doutel, 2017).

En la Tabla tenemos la clasificación de las placas de desarrollo Arduino que podemos encontrar en el mercado:

Tabla 3. Tipos de Arduino.

<b>Nivel de Entrada</b>	Arduino UNO, Arduino PRO, Arduino PRO Mini, Arduino MICRO, Arduino NANO
<b>Características Mejoradas</b>	Arduino MEGA, Arduino ZERO, Arduino DUE
<b>Internet de las cosas</b>	Arduino YUN
<b>Vestible</b>	Arduino GEMMA, Arduino LILYPAD
<b>Impresión 3D</b>	MATERIA 101
Fuente: (Doutel, 2017)	

#### 4.2.2.2. Shields de arduino

Así como las personas a los Arduinos se les puede vestir con accesorios que les otorgan capacidad que antes no tenían, estos accesorios se ponen sobre el Arduino en cuestión (Shields para Arduino, s.f.).

Según Doutel “Gracias a los *shields* podemos contar con infinidad de funciones, como GPS, relojes en tiempo real, conectividad por radio, pantallas táctiles LCD, placas de desarrollo, y un larguísimo etcétera de elementos” (Doutel, 2017).

En la Tabla tenemos otros ejemplos de *shields* que pueden ser usados en Arduino

Tabla 4. *Shields* de Arduino.

<b>Nivel de Entrada</b>	Arduino MOTOR <i>Shield</i>
<b>Características Mejoradas</b>	Arduino PROTO <i>Shield</i>
<b>Internet de las cosas</b>	Arduino ETHERNET <i>Shield</i> , Arduino GSM <i>Shield</i> , Arduino WIFI <i>Shield</i>
Fuente: (Doutel, 2017)	

#### 4.2.3. Computadora

Una computadora es un dispositivo electrónico, utilizado para muchos propósitos que van desde hacer cálculos simples o escribir textos, hasta controlar aviones o más según Joyanes “es utilizado para procesar información y obtener resultados, capaz de ejecutar cálculos y tomar decisiones a velocidades millones o cientos de millones más rápidas que puedan hacerlo los humanos.” (Joyanes Aguilar, 2008).

##### 4.2.3.1. Arquitectura

Muchas han sido los cambios desde la creación de las primeras computadoras, aunque eso no quiera decir que sean diferentes las actuales de las anteriores, dado a que solo ha ido aumentado las capacidades de las diferentes partes que la conforman (Computadora , s.f.).

La conexión mediante buses no ha cambiado, así como la arquitectura básica que sigue siendo:

- Memoria
- Dispositivos de Entrada/Salida
- Unidad Aritmético Lógica
- Unidad de Control

#### **4.2.3.2. Clasificación de las computadoras**

Según Joyanes “Las computadoras modernas se pueden clasificar en

- Computadoras Personales (*PC*)
- Servidores
- Minicomputadora
- Grandes Computadoras
- Supercomputadoras” (Joyanes Aguilar, 2008)

#### **4.2.3.3. Computadora de placa única (sbc)**

A diferencia de una computadora normal de escritorio que tiene que ser armada usando diferentes partes y materiales, desde la caja, tarjeta madre, procesador, etc., o una computadora portátil que tiene el mismo principio de que es armada por partes, una computadora de placa única a su vez ya tiene todas estas partes ensambladas en una sola PCB que normalmente suele ser del tamaño de la palma de una mano (Griffith, 2017).

Una computadora de placa una suele tener lo siguiente:

- “Un procesador de aplicaciones que pueda ejecutar un sistema operativo de alto nivel, como Linux, Android o Windows.
- Funcionalidad lista para usar.
- Conectividad inalámbrica y por cable.
- Un sólido ecosistema de usuarios, tarjetas de expansión y software de código abierto” (Griffith, 2017).

En la Tabla 5 tenemos algunas computadoras de placa única así como algunas de sus características

Tabla 5. Comparación de computadoras de placa única.

	<b>Raspberry Pi 3 B+</b>	<b>BeagleBone Black</b>	<b>Intel Joule 570X</b>	<b>nVidia Jetson TK1</b>	<b>DragonBoard</b>
<b>Procesador</b>	Broadcom BCM2837B0 basado en ARM Cortex-A53	TI Am3358 basado en ARM Cortex-A8	Intel Atom T5700	nVidia 4-plus-1 basado en ARM Cortex-A-15	Qualcomm Snapdragon 410C basado en ARM Cortex-A53
<b>Arquitectura</b>	64 bit RISC	32 bit RISC	64 bit x86	32 bit RISC	64 bit RISC
<b>Núcleos</b>	4 CPU	1 CPU	4 CPU	4 CPU 1 GPU	4 CPU
<b>Velocidad de reloj</b>	1.4 GHz	1 GHz	1.7 GHz, overclocking 2.4 GHz	2.3 GHz	1.2 GHz
<b>GPU</b>	Broadcom VideoCore IV	PowerVR SGX530	Intel HD	nVidia Kepler GPU	Qualcomm Adreno 306
<b>Tipo de Memoria</b>	LPDDR2	DDR3	LPDDR4	DDR3L	LPDDR3
<b>Cantidad de Memoria</b>	1 GB	512 MB	4 GB	2 GB	1 GB
<b>Almacenamiento a Bordo</b>	No	4 GB eMMC Flash	16 GB eMMC Flash	16 GB eMMC Flash	8 GB eMMC Flash
<b>Sistemas Operativos Compatibles</b>	Android, Varias Distribuciones de Linux, Windows 10 IoT Core	Android, Varias Distribuciones de Linux	Ubuntu Desktop/Core/Ostro/Y octo Linux, Windows 10 IoT Core	Linux 4 Tegra (Ubuntu), Android	Android, Varias Distribuciones de Linux, Windows 10 IoT Core
<b>WiFi</b>	A/G/N	No	A/GN/AC	No	B/G/N
<b>Bluetooth</b>	BT4.1	No	BT4.1	No	BT4.1
<b>E/S de Propósito General (GPIO)</b>	40 Posibles con Reasignación	69 Posibles con Reasignación	8 Dedicados/48 al Reasignar Pines de Interfaz	8 Dedicados	12
<b>Voltaje de E/S de Propósito General (GPIO)</b>	3.3V	3.3V	1.8V	1.8V	1.8V
<b>Modulación de Ancho de Pulso (PWM)</b>	2	8	4	4	0

<b>USB</b>	4x Tipo A	1x Tipo A-Mini, 1x Tipo A <i>host</i>	1x Tipo C 3.1, 1x Tipo A 3.0	1x Tipo A-Micro, 1x Tipo A 3.0	1x Tipo A-Micro, 2x Tipo A 2.0
<b>Interconexión de Componente Periférico Express (PCIe)</b>	No	No	1 Puerto - 5GB/s	Media Ranura Mini-PCIe	No
<b>Interfaz de cámara</b>	Si (1xMIPI-CSI)	Si (1xGPMC)	Si (2xMIPI-CSI)	Si (2xMIPI-CSI, x4 y x1)	Si (1xMIPI-CSI)
<b>Salida HDMI</b>	1080p60	1080p24	1080p60	2160p30	1080p30
<b>Interfaz de Pantalla</b>	HDMI	microHDMI	HDMI, MIPI-DSI(4K)	HDMI, LVDS	HDMI, MIPI-DSI
<b>Codificador de video 4K</b>	No	No	4kx2kp60 (H.264), 4kx2kp30(MVC), 1080p60(VP8), 1080p30(VP9)	4kx2kp24 (H.264)	1080p30(H.264)
<b>Decodificador de video 4K</b>	No	No	4kx2kp60(H.264, MVC, VP8, VP9)	4kx2kp30 (H.264)	1080p30(H.264), 720p (H.265)
<b>Consumo de Poder</b>	250-750mA@5V	210-460mA@5V	130-600mA@12V	250-4800mA@12V	50-400mA@12V
Fuente: (Griffith, 2017)					

#### 4.2.4. Sensor

Un sensor como su nombre dice sirve para sensar, es decir tomar las variables físicas y medirlas para luego transformarlas en señales eléctricas que puedan ser entendidas por una computadora o microcontrolador (Pallás Areny, 2004).

Los sensores según Pérez & Gardey “Estos aparatos pueden transformar las magnitudes físicas o químicas en magnitudes eléctricas” (Pérez Porto & Gardey, 2010). La estructura interna de un sensor en general

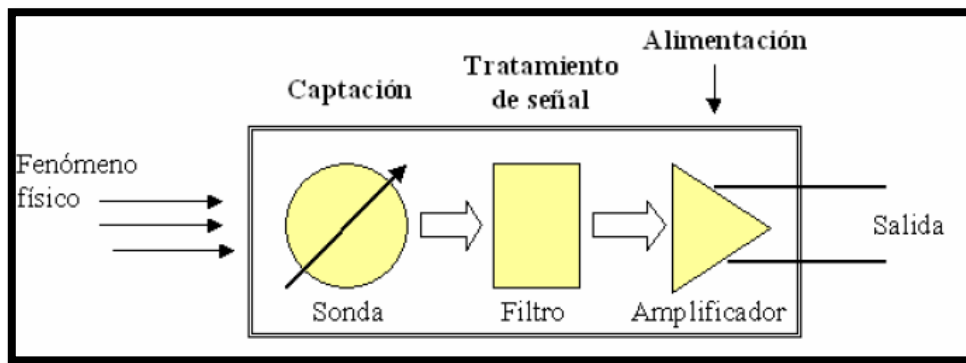


Figura 5. Funcionamiento de un sensor

Fuente: (Sensores para Domótica e Innótica, s.f.)

##### 4.2.4.1. Tipos de sensores

Existe una gran cantidad de sensores y cada uno sirve para medir una o varias unidades de medida con su respectiva magnitud por tanto los sensores se pueden clasificar de diversas formas, según algún criterio o característica como vemos en la Tabla 6

Tabla 6. Clasificación de los sensores

Criterio	Clases	Ejemplos
<b>Aporte de energía</b>	Moduladores	Termistor
	Generadores	Termopar
<b>Señal de salida</b>	Analógicos	Potenciómetro
	Digitales	Codificador de posición
<b>Modo de operación</b>	De deflexión	Acelerómetro de deflexión
	De comparación	Servoacelerómetro

Fuente: (Pallás Areny, 2004)

#### 4.2.5. Actuador

Los actuadores son aparatos que transforman una energía de entrada en una acción como puede ser movimiento. “Usualmente, los actuadores generan una fuerza mecánica a partir de distintos tipos de energía, como puede ser eléctrica, neumática, o hidráulica” (¿Qué son los actuadores electrónicos?, 2013).

##### 4.2.5.1. Tipos de actuadores

En la Tabla 7 podemos observar los tipos de actuadores que existen, así como su forma de producir movimiento entre otras características:

Tabla 7. Clasificación de los actuadores

	<b>Actuador Neumático</b>	<b>Actuador Eléctrico</b>	<b>Actuador Hidráulico</b>
<b>Fuerza Generadora de Movimiento</b>	Presión de aire	Energía eléctrica	Presión hidráulica
<b>Elemento Motriz</b>	Émbolo, Pistón o Veleta	Motor Eléctrico	Émbolo, Pistón o Veleta
<b>Transmisión de Fuerza o Torque</b>	Eje o Cremallera	Reductor	Eje
<b>Conversión Mecánica</b>	Yugo o Piñón	- No hay -	Yugo o Piñón
Fuente: (Actuador, s.f.)			

#### 4.2.6. Sistema operativo y software

##### 4.2.6.1. Sistema operativo

Un Sistema Operativo o como se lo conoce en forma abreviada SO es en palabras simples una interfaz entre el hardware y el software que se instale en el sistema operativo, además que se encarga de que el hardware funcione correctamente valiéndose del software especializado (drivers) (Joyanes Aguilar, 2008).

Los sistemas operativos de hoy en día son diseñados para que las personas normales puedan utilizarlos con cada vez mayor facilidad dado a hace tiempo los sistemas operativos se manejaban a través de los terminales en los que se usan comandos para controlar la computadora, y en la actualidad se hace uso de ventanas, iconos, menús, botones que se encarga el sistema operativo de usar comandos en consecuencia en segundo plano (Sistemas Operativos, s.f.).



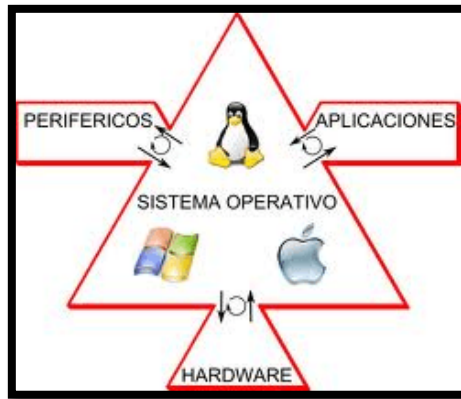


Figura 6. Comunicación de SO con otras partes del sistema

Fuente: (Sistemas Operativos, s.f.)

“Las funciones básicas del Sistema Operativo son administrar los recursos del ordenador, coordinar el hardware y organizar archivos y directorios en los dispositivos de almacenamiento de nuestro ordenador” (Sistemas Operativos, s.f.).

#### 4.2.6.2. Tipos de sistemas operativos

Existen dos tipos de sistemas operativos: SO para PC, estaciones de trabajo, servidores y SO para móviles.

**Sistemas Operativos Para PC.-** “Los más utilizados son el Windows, el Mac y el LINUX” (Sistemas Operativos, s.f.).

**Sistemas Operativos Móviles.-** “Los más utilizados son el Android y el iOS” (Sistemas Operativos, s.f.).



(a)

(b)

Figura 7. Sistemas Operativos para: (a) computadoras y (b) dispositivos móviles

Fuente: (Sistemas Operativos, s.f.)

#### 4.2.7. Software

El software es en sí líneas de comando que le ordenan a la computadora realizar una tarea específica con ayuda del hardware como puede ser el procesador, o usar algún periférico, según Joyanes “Un programa de software es un conjunto de sentencias o instrucciones a la computadora. Existen dos tipos importantes de software: software del sistema y software de aplicación” (Joyanes Aguilar, 2008).

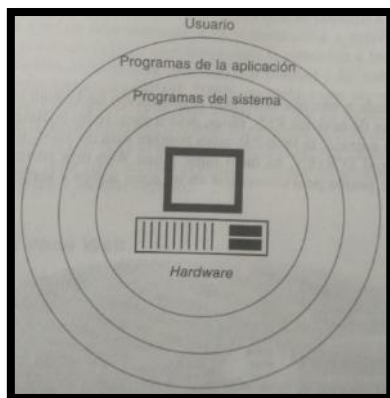


Figura 8. Relación entre programas de aplicación y programas del sistema

Fuente: (Joyanes Aguilar, 2008)

### 4.3. Interfaz web

Así como una interfaz es una agrupación de ventanas, botones, cuadros de texto, etiquetas, etc., las interfaces web no se alejan de esto dado que tienen estos elementos y logran mediante las diferentes herramientas para desarrollo de páginas e interfaces web. Según Milan “En definitiva, la interfaz web es el conjunto gráfico que permite la presentación y la navegación del sitio. Esto se consigue con la inclusión de elementos comunes a toda la web que son estándares, haciendo que los usuarios tengan completo control sobre las funcionalidades del sitio desde el momento mismo de entrar a él sin que para ello deba tener amplios conocimientos ni preparación anterior alguna.” (Milán, 2018)

#### 4.3.1. Componentes

##### 4.3.1.1. Elementos de identificación:

Según Barberá “Son los elementos que identifican un sitio web. El usuario al verlos sabe a qué sitio a accedido. Ejemplos de estos elementos son: logos, título de la página, nombre de dominio, banderas, etc.” (Barberá, s.f.)

#### 4.3.1.2. Elementos de navegación:

Son aquellos que como su nombre indica nos permite desplazarnos por los diferentes elementos de la interfaz web, según Barberá “Habitualmente los elementos de navegación de un sitio se sitúan en la barra de navegación superior y/o en una barra de navegación lateral. El logo del sitio web nos traslada directamente a la página web de inicio. Por lo que el logo es un componente de identificación y navegación.” (Barberá, s.f.)

#### 4.3.1.3. Elementos de contenidos:

Según Barberá “Son las zonas donde se muestra el contenido o información relevante de cada página web de nuestro sitio. Se subdivide en título y contenido propiamente dicho.” (Barberá, s.f.)

#### 4.3.1.4. Elementos de interacción:

“Son las partes de la web en la que se ofrece interactuar al usuario. Por ejemplo: una caja de búsqueda, un selector de idiomas, un formulario de contacto, etc.” (Barberá, s.f.)

### 4.3.2. Factores de diseño

Los factores de diseño más importantes para las interfaces web son:

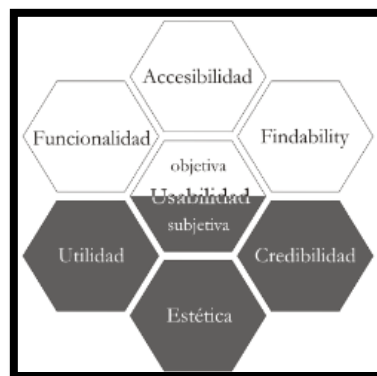


Figura 9. Factores de diseño orientado a la satisfacción-no frustración de uso

Fuente: (Hassan Montero, 2006)

#### 4.3.2.1. Accesibilidad

Siempre resulta engorroso el querer acceder a un página y esta no pueda ser cargada o si esta se torne lenta al ingresar debido al limitado número de conexiones simultaneas este es el primer punto de los factores de diseño, según Hassan & Fernández “Es un atributo de calidad que se refiere a la posibilidad de que el sitio web pueda ser accedido y usado

por el mayor número posible de personas, indiferentemente de las limitaciones propias del usuario o de las derivadas del contexto de uso” (Hassan & Martín Fernández, 2004).

#### **4.3.2.2. Funcionalidad**

Según Hassan “El concepto de funcionalidad se refiere al correcto funcionamiento técnico del sitio web. Al margen de que un sitio –como por ejemplo una aplicación webmail sea accesible y fácil de usar, si no lleva a cabo su función sin errores como hacer llegar un email a su destinatario– no resultará de valor alguno para el usuario” (Hassan Montero, 2006).

#### **4.3.2.3. Findability (encontrabilidad)**

Facilidad y rapidez con la que se puede encontrar en la página aquello que busco desde un principio (García Alcacio, 2017).

Según Rosenfeld & Morville “Es un factor que se refiere a la posibilidad de encontrar o recuperar fácilmente la información necesitada, resultado de una correcta arquitectura de información estructuración, descripción y clasificación de contenidos” (Rosenfeld & Morville, 2002).

#### **4.3.2.4. Utilidad**

La utilidad varía en medida de lo que el usuario está buscando, según Hassan “es la medida en la que el sitio web sirve al usuario provecho, beneficio e interés que le produce– que podemos definir como una relación de actitud afectiva con el sitio web. Hay que aclarar que este factor se refiere a la utilidad subjetiva o percibida, no a la objetiva o técnica (Hassan Montero, 2006).

#### **4.3.2.5. Estética**

El diseño debe ser tomado en cuenta en la creación de la interfaz web aunque es subjetiva dado a que lo que se define como bello, atractivo cambia de usuario a usuario, según Laviera & Tractinsky “ésta cualidad se refiere a una apariencia bella, placentera y atractiva. Un diseño estético es aquel que resulta agradable, ya sea para los sentidos, para la imaginación, o para nuestro entendimiento” (Laviera & Tractinsky, 2004).

#### 4.3.2.6. Credibilidad

“Si bien, al igual que la utilidad, no se trata de un factor puramente de diseño, ya que también depende de variables externas a éste, el diseño es un elemento fundamental en la proyección de confianza” (Hassan Montero, 2006).

#### 4.3.2.7. Usabilidad

No debe confundirse con alguno de los otros factores de diseño dado que la usabilidad según Hassan “se basan en la enumeración de los distintos atributos que la componen y a través de los cuales pretende ser medida o evaluada. Esto se debe a que la ingeniería de la usabilidad se compone en su mayoría de metodologías y herramientas de evaluación, siendo la validación de las decisiones de diseño desde la perspectiva del usuario la función principal de la práctica de la usabilidad” (Hassan Montero, 2006).

### 4.4. Hosting y almacenamiento de información

#### 4.4.1. *Hosting* o alojamiento web

Una definición rápida y fácil es según López “Un *hosting* web es un servicio donde un proveedor te alquila un servidor conectado a Internet” (López, 2018).



Figura 10. Una sala de servidores típica con cientos de máquinas.

Fuente: (López, 2018)

Una de las muchas funcionalidades que provee un hosting según López es “para crear sitios web (que son, más que nada muchos ficheros en formato HTML), y también se los usa simplemente para permitir descargar cualquier cosa “(documentos PDF, ficheros MP3 de audio, vídeo, etc.)” (López, 2018).

#### 4.4.1.1. Tipos de hosting

“Una de las muchas formas de categorizar a los hosting es en función de cómo se organizan dentro del servidor web” (Alonso, 2018):

- “Hosting compartido
- Hosting dedicado
- Hosting elástico
- VPS o Servidor Virtual Privado
- *Hosting* en la nube” (Alonso, 2018)

#### 4.4.1.2. Host ddns

Dynamic DNS o simplemente DDNS es un servicio prestado por una empresa a las personas necesitan crear un nombre dominio y que cuentan con servicio de internet pero no contratado una IP pública estática, DDNS se encarga de transformar un nombre de dominio que es creado dentro de servicio en la IP pública que el usuario tenga actualmente, típicamente es un programa de computadora o actualmente se lo puede configurar en un router, que se encarga de revisar periódicamente si dicha IP ha cambiado y actuar en consecuencia. (DNS dinámicos (DynDNS): accesos remotos sin complicaciones, 2017).

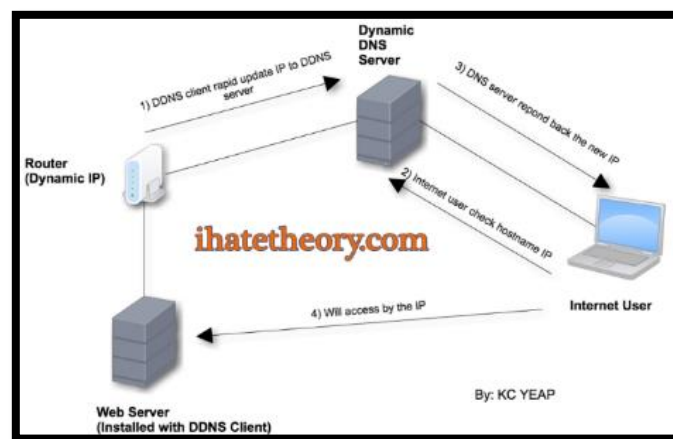


Figura 11. Funcionamiento del DDNS

Fuente: (Dynamic DNS – The automated rapid record updates, 2016)

## 4.4.2. Sql

### 4.4.2.1. Base de datos

Una base de datos son datos que están relacionados, agrupadas o estructuradas. Para permitirnos acceder a aquellos datos se han creado muchas aplicaciones (creadas usando diferentes lenguajes de programación) que de una forma u otra nos permiten actualizar, recuperar o borrar la información guardada. Las bases de datos se forman con una o más tablas que guarda información repartida en filas y columnas según Pérez “las columnas guardan una parte de la información sobre cada elemento que queramos guardar en la tabla, cada fila de la tabla conforma un registro” (Pérez Valdés, 2007).

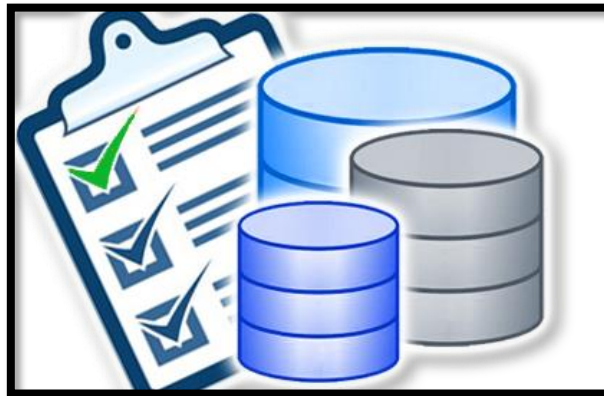


Figura 12. Representación gráfica de una base de datos

Fuente: (Base de Datos (lógica), 2015)

### 4.4.2.2. Lenguaje del sql

Según Plasencia “SQL (Structured Query Language) es un lenguaje... de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en ellas, gracias a la utilización del álgebra y de cálculos relacionales, el SQL brinda la posibilidad de realizar consultas con el objetivo de recuperar información de las bases de datos de manera sencilla. Las consultas toman la forma de un lenguaje de comandos que permite seleccionar, insertar, actualizar, averiguar la ubicación de los datos, y más.” (Plasencia Prado, s.f.).

### 4.4.2.3. Sistemas de gestores de bases de datos (sgbd) comerciales y libres

Son programas que de manera discreta administran y controlan información contenida en una base de datos, dado a que se las realiza para el usuario final están creadas para facilitar

el manejo con el uso de botones, iconos, tablas, etc. (¿Qué es un gestor de datos y para qué sirve?, 2016).

Entre los sistemas de gestores de bases de datos existen tanto los comerciales o propietarios y los que son de software libre, cada uno con sus ventajas y desventajas, como son sus continuas actualizaciones, arreglo de errores y soporte.

#### 4.4.2.4. Sistemas de gestores de bases de datos comerciales

Entre los sistemas de gestores de bases de datos comerciales más usados tenemos:

- **Oracle** “Es un sistema de gestión de base de datos relacional (o RDBMS por el acrónimo en inglés de Relational Data Base Management System), fabricado por Oracle Corporation. (SGBD Comerciales, s.f.). Según Marín “tradicionalmente Oracle ha sido el SGBS por excelencia, considerado siempre como el más completo y robusto, destacando por: su soporte de transacciones, estabilidad, escalabilidad, y ser multiplataforma” (Marín, 2019).
- ◆ **SQL Server:** “En contraposición y como competencia directa a Oracle, está SQL Server de Microsoft. Este se ejecuta en Transact-SQL, un conjunto de aplicaciones que agregan funciones al programa, como el manejo de errores y excepciones, manejo de datos, sacar de información directo de Web, así como usar otros lenguajes de programación, etc.” (Los mejores gestores de base de datos del mercado, 2017).

Según Marín “sus principales características son:

- Soporte exclusivo por parte de Microsoft.
- Escalabilidad, estabilidad y seguridad.
- Posibilidad de cancelar consultas.
- Potente entorno gráfico de administración que permite utilizar comandos DDL y DML.
- Aunque es nativo para Windows puede utilizarse desde hace ya un tiempo en otras plataformas como Linux o Docker” (Marín, 2019)
- Uniones hash (Marín, 2019)



#### 4.4.2.5. Sistemas de gestores de bases de datos libres

Entre los sistemas de gestores de bases de datos libres más usados tenemos:

- ◆ **MySQL:** MySQL es un gestor de bases de datos relacionales rápido, sólido y flexible. Está desarrollado en C y C++, lo que ayuda a su integración con otros programas que usen esos lenguajes (Sistema Gestor de Base de Datos , s.f.).

Es también multihilo y multiusuario y es el más usado en aplicaciones open software, dado que tiene licencia GNU GPL, pero, los que quieran usarlo en productos privados deben adquirir una licencia.

Ventajas:

- Velocidad al realizar las operaciones
  - Bajo costo en requerimientos para la elaboración de bases de datos
  - Facilidad de configuración e instalación (Marín, 2019).
- ◆ **PostgreSQL:** Es un sistema de gestión de base de datos relacional orientada a objetos y libre. El desarrollo de PostgreSQL es manejado por una comunidad de desarrolladores de manera altruista, libre y/o apoyada por organizaciones comerciales. La comunidad PostgreSQL se denominada el PGDG (PostgreSQL Global Development Group).

Sus principales características son:

- Alta concurrencia: mediante un sistema denominado MVCC (Acceso concurrente multiversión, por sus siglas en inglés)
  - Amplia variedad de tipos nativos: provee nativamente varios soportes
  - Ahorros considerables de costos de operación
  - Estabilidad y confiabilidad (Marín, 2019)
- ◆ **FireBird:** “Potente y, a la vez, sencillo sistema de gestión de base de datos relacional SQL. Es uno de los mejores gestores de código abierto (Open Source) compatible con Windows y Linux. Entre otras funciones, da soporte completo para procedimientos almacenados, transacciones compatibles con las características ACID y métodos de acceso múltiple (nativo, Python, .NET, etc...)” (Los mejores gestores de base de datos del mercado, 2017).

## 5. MATERIALES Y MÉTODOS

### 5.1. Materiales

Para la elaboración del presente proyecto de titulación se utilizaron los siguientes materiales y equipos:

#### 5.1.1. Materiales bibliotecarios y de oficina

Se usaron: libros, internet, computador portátil, impresora, entre otros.

##### 5.1.1.1. Materiales

Tabla 8. Materiales utilizados para el desarrollo del proyecto.

<b>MATERIAL</b>	<b>UNIDAD</b>	<b>CANTIDAD</b>
Armazón metálica	Unidad	1
Cajón de Madera	Unidad	1
Tierra	Saco	3
Agua	Unidad	Ilimitado
Manguera	Metros	4
Semilla de Lechuga	Funda	3
Balde	Unidad	1
Cautín	Unidad	1
Estaño	Metro	1
Pasta para soldar	Unidad	1
Fuente: [El autor]		

Tabla 9. Equipos utilizados para el desarrollo del proyecto

<b>MATERIAL</b>	<b>UNIDAD</b>	<b>Cantidad</b>
Arduino MEGA 2560	Unidad	1
Arduino NANO	Unidad	1
Raspberry Pi 3	Unidad	1

<b>Motores Pololu Nema 23</b>	<b>Unidad</b>	<b>4</b>
<b>Drivers DRV8825</b>	<b>Unidad</b>	<b>4</b>
<b>Motores Paso a Paso BYJ-24</b>	<b>Unidad</b>	<b>3</b>
<b>Drivers ULN2003</b>	<b>Unidad</b>	<b>3</b>
<b>Motores DC Sankyo SHU9L</b>	<b>Unidad</b>	<b>3</b>
<b>Drivers L298N</b>	<b>Unidad</b>	<b>2</b>
<b>Mini Motores Vibradores</b>	<b>Unidad</b>	<b>3</b>
<b>Sensor de Temperatura/Humedad DHT-22</b>	<b>Unidad</b>	<b>1</b>
<b>Sensor de Humedad de Suelo YL-38</b>	<b>Unidad</b>	<b>1</b>
<b>Sensor de Nivel de Agua</b>	<b>Unidad</b>	<b>1</b>
<b>Bomba de Agua</b>	<b>Unidad</b>	<b>1</b>
<b>Electroválvula</b>	<b>Unidad</b>	<b>1</b>
<b>Fuente de Poder DC 12V</b>	<b>Unidad</b>	<b>4</b>
<b>Fuente: [El Autor]</b>		

## 5.2. Métodos

### 5.2.1. Especies vegetales del cantón Loja

Se ha realizado una investigación a fin de encontrar los diferentes tipos de plantas que se siembran y cosechan en el cantón Loja de la provincia de Loja a fin de escoger posteriormente las plantas más óptimas para su uso en el prototipo para siembra automatizado, para lo cual se ha desarrollado la siguiente tabla:

Tabla 10. Especies vegetales del cantón Loja

	<b>Especie vegetal</b>			
<b>Cantón Loja</b>	<ul style="list-style-type: none"> <li>• Maíz</li> <li>• Maní</li> <li>• Fréjol</li> <li>• Arveja</li> <li>• Haba</li> <li>• Lenteja</li> <li>• Vainita</li> <li>• Zarandaja</li> <li>• Trigo</li> </ul>	<ul style="list-style-type: none"> <li>• Camote</li> <li>• Cebolla</li> <li>• Culantro</li> <li>• Perejil</li> <li>• Orégano</li> <li>• Apio</li> <li>• Ajo</li> <li>• Ají</li> <li>• Verdolaga</li> </ul>	<ul style="list-style-type: none"> <li>• Cola de caballo</li> <li>• Caña de azúcar</li> <li>• Guaba</li> <li>• Manzana</li> <li>• Cidra</li> <li>• Pera</li> <li>• Mandarina</li> </ul>	<ul style="list-style-type: none"> <li>• Capulí</li> <li>• Mango</li> <li>• Fresa</li> <li>• Mora</li> <li>• Tuna</li> <li>• Zapallo</li> <li>• Zambo</li> <li>• Zuquini</li> <li>• Zapote</li> </ul>

<ul style="list-style-type: none"> <li>• Cebada</li> <li>• Zanahoria</li> <li>• Remolacha</li> <li>• Melloco</li> <li>• Coliflor</li> <li>• Nabo</li> <li>• Acelga</li> <li>• Yuca</li> <li>• Papa</li> <li>• Guineo</li> <li>• Plátano</li> <li>• Maguey</li> <li>• Col</li> <li>• Lechuga</li> <li>• Rábano</li> <li>• Brócoli</li> </ul>	<ul style="list-style-type: none"> <li>• Espinaca</li> <li>• Condurango</li> <li>• Jazmín</li> <li>• Tilo</li> <li>• Sauco</li> <li>• Higo</li> <li>• Tomate</li> <li>• Papaya</li> <li>• Tomate de árbol</li> <li>• Babaco</li> <li>• Tumbo</li> <li>• Luma</li> <li>• Jícama</li> <li>• Tabaco</li> <li>• Níspero</li> </ul>	<ul style="list-style-type: none"> <li>• Naranja</li> <li>• Toronja</li> <li>• Lima</li> <li>• Limón</li> <li>• Naranja</li> <li>• Maracuyá</li> <li>• Guayaba</li> <li>• Sandía</li> <li>• Aguacate</li> <li>• Chirimoya</li> <li>• Granadilla</li> <li>• Piña</li> <li>• Uva</li> <li>• Reina claudia</li> <li>• Ciruela</li> <li>• Café</li> </ul>	<ul style="list-style-type: none"> <li>• Ahojcha</li> <li>• Pepino</li> <li>• Pimiento</li> <li>• Toronjil</li> <li>• Menta</li> <li>• Llantén</li> <li>• Malva</li> <li>• Cedrón</li> <li>• Hierba luisa</li> <li>• Manzanilla</li> <li>• Borraja</li> <li>• Cacao</li> <li>• Orquídea</li> </ul>	
Fuente: [El Autor]				

Dado que el período de crecimiento de ciertas plantas es muy extenso hasta la obtención de plántulas por ejemplo plantas frutales, se resolvió usar aquellas que se desarrollan con mayor rapidez por ejemplo hortalizas más concretamente las siguientes: lechuga, col.

### 5.2.2. Diseño del armazon del sistema robótico automatico

Para el diseño del sistema robótico se basó en un robot cartesiano, es decir, el uso de tres ejes X, Y y Z por lo cual el sistema robótico se moverá de manera lineal, por lo tanto, se consideró en el diseño dos sistemas de transmisión de movimiento:

- Movimiento mediante correas
- Movimiento mediante pernos

Se decidió el uso de transmisión del movimiento mediante pernos que nos proporciona un movimiento más fluido y una menor probabilidad de perdida de pasos.

Para el diseño se usó el software para diseño 3D, con lo cual el sistema robótico se fue desarrollando en piezas para posteriormente unir las y formar el armazón final; se empezó con la base quedando de la siguiente manera:

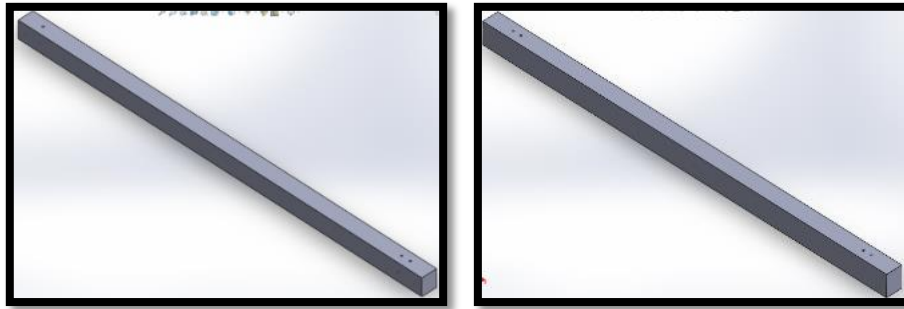
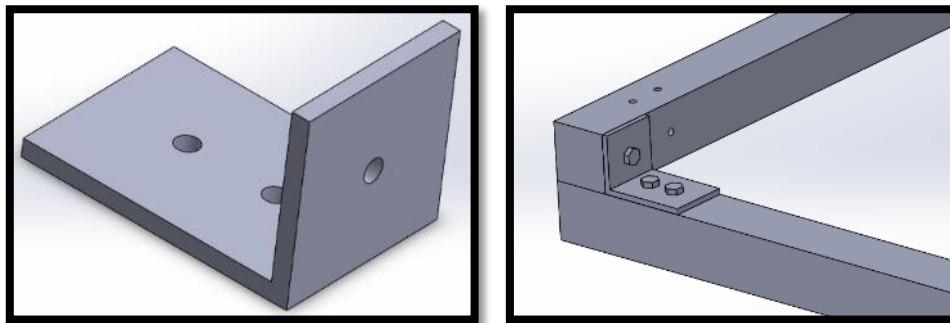


Figura 13. Barra para base del armazón

Fuente: [El Autor]



(a)

(b)

Figura 14. a) Unión para las barras de la base; b) Barras unidas

Fuente: [El Autor]

Empezando con la barra inferior del eje x, seguida de la barra transversal, una unión en forma de L, quedando estas piezas unidas con ayuda de unos pernos, tal como que vemos en la figura anterior, y finalmente utilizando dos de cada barra y 4 uniones nuestra base nos queda:

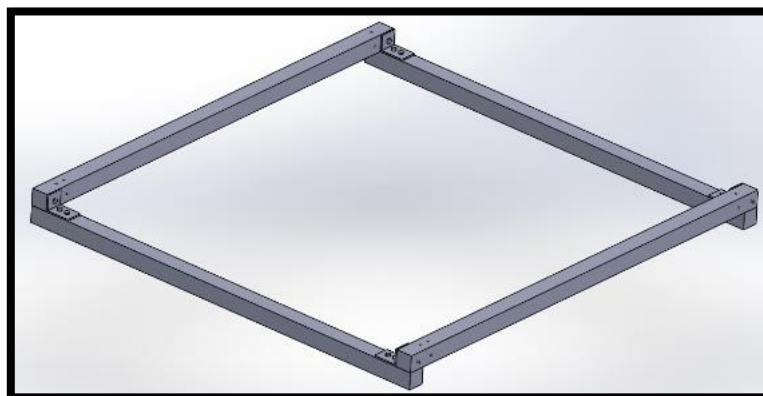


Figura 15. Base del armazón terminado

Fuente: [El Autor]

Para la colocación de los motores se utilizó unas platinas soldadas de la forma en la que vemos en la figura, y en contraparte con la que platina que soporta al perno y barra calibrada, dichos soporte se ubican sobre la barra inferior del eje x quedando como vemos en la figura

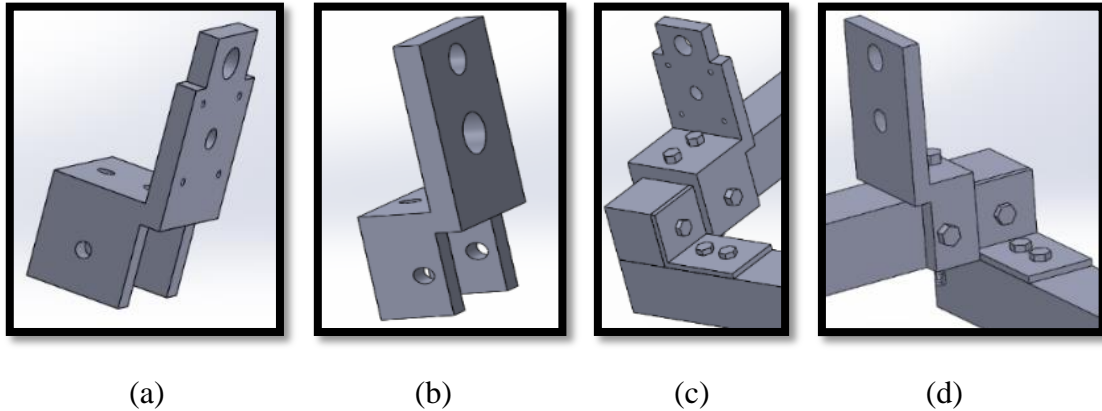


Figura 16. a) Soporte para motor y barra calibrada; b) Soporte para perno y barra calibrada; c) Soporte de motor y barra ubicado en el armazón; d) Soporte de perno y barra ubicado en el armazón

Fuente: [El Autor]

Quedando todos los soportes ubicados de la siguiente manera:

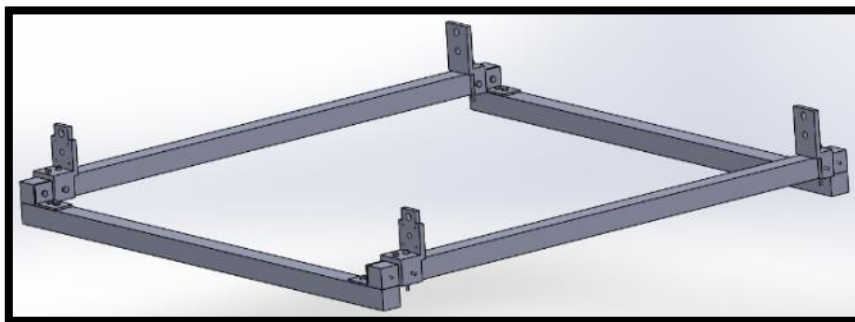
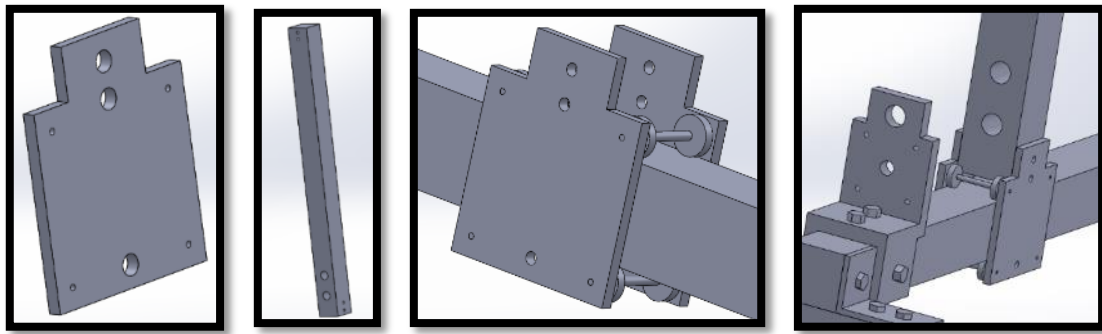


Figura 17. Base del armazón con soportes de motores, pernos y barras calibradas

Fuente: [El Autor]

Para realizar el movimiento en el eje x se utilizó un perno de  $\frac{3}{4}$ ", una tuerca con la respectiva medida para el perno soldada a la barra de subida al eje y, se usó también unos soportes en los que se ubicó ruedas para ayudar a aminorar la carga sobre el perno y facilitar el movimiento, así como una barra calibrada como soporte y seguridad hacia el eje y



(a)

(b)

(c)

(d)

Figura 18. a) Soporte de ruedas y subiente; b) Barra subiente eje y; c) Soporte con ruedas ubicadas en base del armazón; d) Soporte con ruedas y barra de subida ubicado en armazón

Fuente: [El Autor]

El armazón finalmente nos queda de la siguiente manera:

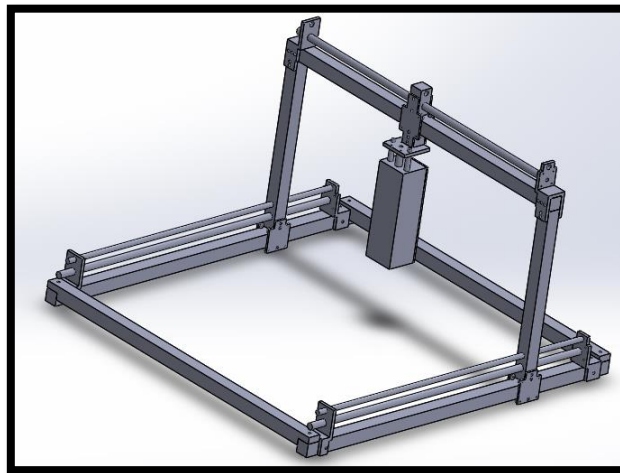


Figura 19. Diseño del sistema robótico, en software de modelado 3D

Fuente: [El Autor]

Dicho diseño se consultó con un ingeniero para verificar que funcione como estaba planeado; luego se procedió a la consultar al técnico del taller mecánico el material más óptimo para la construcción del armazón del sistema robótico, y se optó por el uso de metal; se usó tubo cuadrado para la base, los laterales y superior, y platinas para unir los tubos, soporte para las ruedas y para soporte de motor, perno y barras calibradas.



(a)



(b)

Figura 20. (a) Base del armazón; (b) Platinas de unión

Fuente: [El Autor]



(a)



(b)

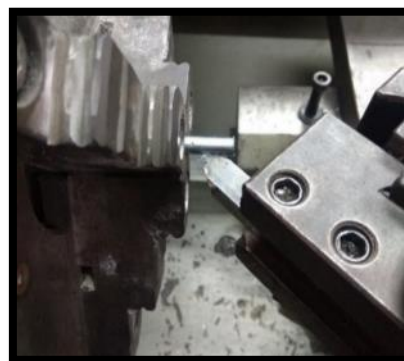
Figura 21. (a) Platinas para soporte de motores, pernos y barras calibradas; (b) Platinas de soporte *para* *ruedas*

Fuente: [El Autor]

Además del uso del perno para el movimiento se usó también ruedas para disminuir el esfuerzo del motor.



(a)



(b)

Figura 22. (a) Ejes para las ruedas; (b) Torneado de los ejes de las ruedas

Fuente: [El Autor]



Quedando las platinas los ejes y las ruedas de la siguiente manera:

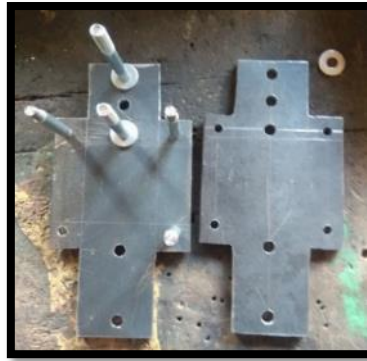


Figura 23. Platina, ejes y ruedas

Fuente: [El Autor]

Algunos de los soportes para los motores, y soporte para las ruedas, ya ubicados quedaron de la siguiente manera:



(a)



(b)

Figura 24. (a) Soporte para motores; (b) Soporte para ruedas

Fuente: [El Autor]

Se procedió a crear un surco en la barra calibrada, con ayuda de un torno para poder ubicarle anillos de seguridad para que la barra calibrada no se mueva de su posición



Figura 25. Barra calibrada con anillo de seguridad

Fuente: [El Autor]

Con respecto a los pernos se les procedió a torneado uno de sus lados para poder ubicar la una brida que va conectada a su par que se encuentra en el eje del motor



(a)



(b)

Figura 26. (a) Perno antes de ser torneado; (b) Perno luego de ser torneado

Fuente: [El Autor]

Una vez terminado, se procedió al armado, ubicación de motores y al pintado del armazón, quedando de la siguiente manera:

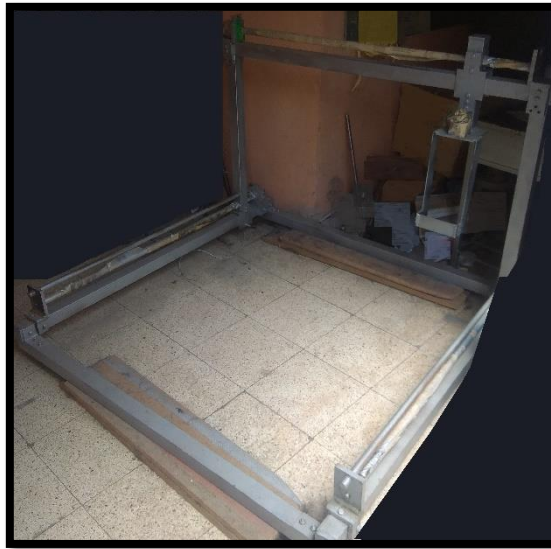


Figura 27. Armazón terminado

Fuente: [El Autor]

### 5.2.3. Actuadores

En el sistema robótico automático se van a utilizar diferentes actuadores, para distintos fines; dado que se consideró en el diseño un sistema cartesiano, para el movimiento general del sistema robótico en los ejes X, Y y Z se va a usar motores paso a paso debido a que nos permite comparado con un motor dc normal un control preciso de los movimientos debido su forma de construcción gira una cantidad de grados (pasos) dependiendo de la entrada de control, y comparado con un servomotor gira 360° continuamente como un motor dc normal.

Para la elección de los motores paso a paso, se realizaron distintos cálculos con el fin de encontrar el torque necesario para mover el armazón del sistema robótico sin mucho esfuerzo o desperdiciando mucha potencia, a continuación, se muestran las distintas fórmulas empleadas y una tabla que nos muestra un resumen de los cálculos.

#### Calculo de Torque

Formulas:

$J_T \rightarrow$  Inercia Total

$$J_T = J_{Motor} + J_{Reductor} + \left( (J_{Acoplamiento} + J_{Tornillo} + J_W) / i \right) \quad (Ec. 1)$$

$J_W \rightarrow$  Inercia Peso

$$J_W = (W/e) * (1/2\pi P)^2 \quad (Ec. 2)$$

$$d_{carga} = distancia * rotación \quad (Ec. 3)$$

$$promedio = (0,37 + 0,3 + 0,3 + 0,36)/4 = 0,334 \text{ cm}$$

$$d_{carga} = 0,334 \text{ cm} * \frac{10 \text{ mm}}{rev * cm} = 3,34 \text{ mm/rev}$$

$$P = pitch = \left(1/d_{carga}\right) \quad (Ec. 4)$$

$$P = \frac{1}{d_{carga}} = \frac{1}{3,34 \text{ mm/rev}}$$

$$P = 0,2994 \text{ rev/mm} = 299,4 \text{ rev/m}$$

$J_{Tornillo} \rightarrow$  Inercia Tornillo

$$J_{Tornillo} = (\pi * L * \rho * r^4)/2 \quad (Ec. 5)$$

$J_{(Tornillo+carga)/Motor} \rightarrow$  Inercia de Tornillo + Carga

$$J_{(Tornillo+carga)/Motor} [Kg - m^2] = (J_{Tornillo} + J_W)/i^2 \quad (Ec. 6) \quad \rightarrow \quad i = 1$$

Perfil de movimiento

$P_{Total} \rightarrow$  Pulsos totales

$$P_{Total} = \left( \frac{D_{Total}}{[d_{carga}/i]} \right) * \theta_{paso} \quad (Ec. 7)$$

$$P_{Total} = 600 \text{ pulsos}$$

$f_{Trapez} \rightarrow$  Frecuencia trapezoidal

$$f_{Trapez} = [P_{Total} - (f_o * t_{ace})]/[t_{Total} - t_{ace}] \quad (Ec. 8)$$

$$f_{Trapez_{M1,M2,M3,M4}} = 266,67 \text{ RPM} \approx 267 \text{ RPM}$$

$T_{movimiento} \rightarrow$  Torque de Movimiento

$$T_{movimiento} = T_{ace} + T_{resis} \quad (Ec. 9)$$

$T_{ace}$  → Torque de Aceleración

$$T_{ace} = (J_{Total}[Kg - m^2]) * \left( \frac{\Delta_{velocidad}[RPM]}{\Delta_{tiempo}[s]} \right) * \frac{2\pi}{60} [N - m] \quad (Ec. 10)$$

$T_{Resis}$  → Torque de Resistencia

$$T_{Resis} = \left( \left( \frac{F_{Total}}{(2\pi P)} \right) + T_{preload} \right) / i \quad (Ec. 11)$$

$F_{Total}$  → Fuerza Total

$$F_{Total} = F_{ext} + F_{friccion} + F_{gravedad} \quad (Ec. 12)$$

$F_{friccion}$  → Fuerza de Fricción

$$F_{friccion} = \mu * Peso * \cos(\theta) * 9,81 \quad (Ec. 13)$$

$F_{gravedad}$  → Fuerza de Gravedad

$$F_{gravedad} = Peso * \sin(\theta) * 9,81 \quad (Ec. 14)$$

Tabla 11. Calculo de torque para elección de motor paso a paso

	Motor X-1	Motor X-2	Motor Y	Motor Z
$J_w [\mu Kg - m^2]$	4,71	4,71	3,14	0,942
$L \rightarrow$ Longitud – Tornillo[m]	1,2	1,2	1,2	0,7
$\rho \rightarrow$ densidad [ $Kg/m^3$ ]	7700	7700	7700	7700
$r \rightarrow$ radio [m]	0,0127	0,0127	0,0127	0,0127
$J_{Tornillo} [\mu Kg - m^2]$	377,58	377,58	377,58	220,25
$J_{(Tornillo+carga)/Motor}$ [ $\mu Kg - m^2$ ]	381,29	381,29	380,72	221,192
$T_{ace} [N - m]$	0,01425	0,01425	0,01419	0,00825
$F_{friccion} [N]$	22,07	22,07	14,72	4,42
$F_{gravedad} [N]$	0	0	0	29,43
$F_{Total} [N]$	22,07	22,07	14,72	33,85
$T_{Resis} [N - m]$	0,01173	0,01173	0,00782	0,01799

$T_{movimiento} [N - m]$	0,02598	0,02598	0,02201	0,02624
Fuente: [El Autor]				

Con los resultados de los cálculos los motores paso a paso que se utilizará serán unos Nema 23 de Pololu híbridos de 200 Pasos/Revolución, que funcionan a 5.7V, 1 A/Fase y cuenta con 0,392266 N-m de torque, que superan el torque necesario según los cálculos y nos dan un buen margen de error.



Figura 28. Motor pasa a paso Nema 23 de Pololu

Fuente: (Stepper Motor, s.f.)

El voltaje indicado de funcionamiento del motor, es decir 5,7V, es un voltaje nominal lo que nos permite usar un voltaje más alto al antes expuesto sin que esto afecte negativamente a nuestro motor siempre y cuando se respete el límite de corriente máxima que soporta este, también debido a que los drivers DRV8825 (apartado 5.2.5.2) utilizados para controlar los motores necesitan un nivel de voltaje mínimo de 8,2V para su funcionamiento y debido a que este voltaje es diferente al que nos puede proveer una fuente comercial como por ejemplo una fuente de 5V o 12V, se procedió a la creación de un placa PCB que se encarga de regular el voltaje de la fuente comercial y llevarla a un nivel aceptable para el funcionamiento conjunto del driver y el motor Nema 23, los reguladores de la placa PCB se ajustaran a 8,2V el voltaje mínimo para activar los drivers.

La placa PCB se la diseño con el software Proteus en su versión 8.1, con la herramienta ISIS se procedió a seleccionar todos los dispositivos que se iban a utilizar en el circuito, y después se procedió a conectar los elementos entre sí; después se utilizó la herramienta ARES para construir la placa ubicando los elementos que se seleccionaron anteriormente y seleccionando el tamaño del líneas de cobre, luego se procedió a imprimir en un papel especial, el circuito en ARES quedo de la siguiente manera.

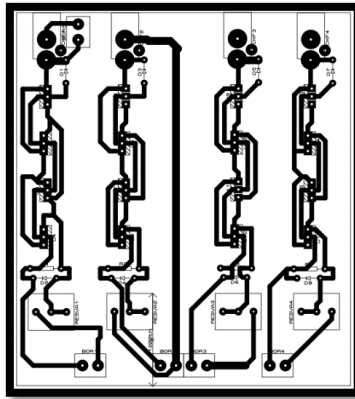


Figura 29. Placa PCB creada en ARES del software Proteus

Fuente: [El Autor]

Con el circuito impreso se procedió a transportar del circuito impreso a la placa de baquelita, posteriormente se procedió a usar ácido para eliminar el exceso de cobre en la baquelita y después se procedió a perforar la baquelita para ubicar los diferentes elementos y luego soldarlos con ayuda de un caudín y estaño, quedando de la siguiente manera:



Figura 30. Placa PCB final

Fuente: [El Autor]

Para el **dosificador de semillas** se va a usar motores dc normales, dado que el dosificador diseñado solo necesita un movimiento circular uniforme que se puede suministrar con un motor dc normal y se puede controlar su velocidad con el uso de *PWM* (*pulse-width modulation*, modulación por ancho de pulsos). A los motores se les ubico una brida circular con un hueco cerca de uno de sus bordes, sobre esta brida y motor se ubicó un contenedor para las semillas con un orificio en su parte inferior, Cuando ambos orificios convergen la semilla cae.



Figura 31. Motor dc

Fuente: (Wagner, s.f.)

Junto con los motores DC se utilizó motores vibradores para la creación del dosificador de semillas, estos se encargaran de que las semillas ubicadas en los contenedores de los dosificadores caigan con mayor facilidad hacia el surco de tierra cuando se inicie el proceso de siembra de semillas.



Figura 32. Motor Vibrador

Fuente: (Vibrador mini Motor, s.f.)

En el interior de la caja se necesita subir y bajar los distintos elementos que se encuentran dentro de ella, por lo cual se va a hacer uso de motores paso a paso para este fin.

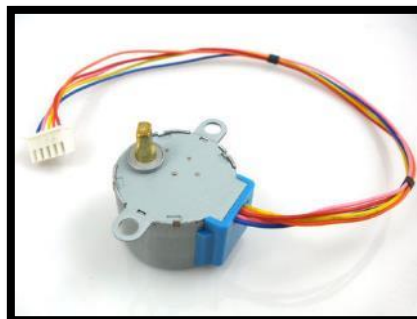


Figura 33. Motor pasó a paso

Fuente: (Geeetech, s.f.)



#### 5.2.4. Equipos de monitoreo de variables físicas

Para el monitoreo de variables físicas se decidió el uso de distintos sensores que permiten una fácil y rápida conexión a un equipo de control, los sensores que se usaran son:

##### 5.2.4.1. Dht22

Es un sensor tanto de humedad como de temperatura ambiental bastante preciso en sus mediciones y es fácil de usar dado que es compatible con cualquier microcontrolador, por lo que se lo puede conectar a un Arduino para obtener la medición de las variables mencionadas para su posterior implementación en una base de datos.

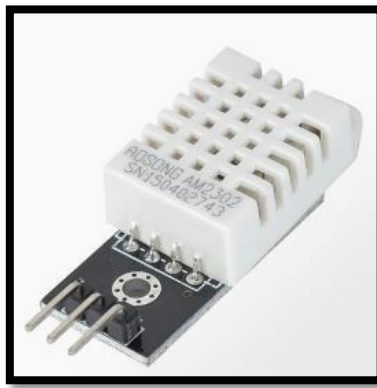


Figura 34. Sensor de temperatura y humedad ambiental DHT22

Fuente: (Sensor de Temperatura y Humedad DHT22, s.f.)

##### 5.2.4.2. YL-38

Es un sensor que sirve para medir la humedad de la tierra, cuenta con dos electrodos en forma de bidente y un circuito de control que se conecta a los electrodos y a un microcontrolador, por tanto al Arduino y de este a la base de datos.

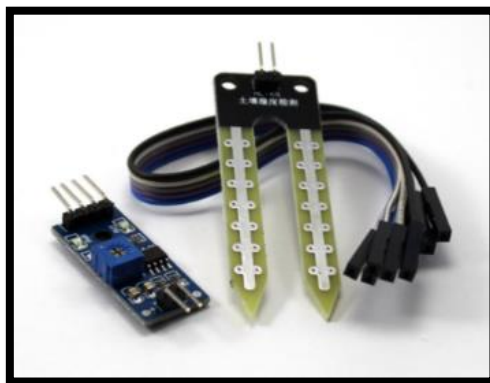


Figura 35. Sensor de humedad de tierra YL-38

Fuente: (Sensor de humedad del suelo YL38, s.f.)

## 5.2.5. Equipos de control

### 5.2.5.1. Raspberry pi

Para el control principal del sistema automatizado se decidió el uso de una computadora de placa única, que viene siendo lo mismo que una computadora de escritorio o una computadora portátil pero de un tamaño muy reducido (aproximadamente el tamaño de una mano), si bien no cuenta con una potencia de procesamiento muy alta, ni mucha memoria *RAM* como estos últimos, nos da las mismas ventajas que estos, y nos permite la instalación de un sistema operativo y uso de software adaptado para estas computadoras de placa única. En el medio se tiene a disposición algunas marcas y modelos de computadora de placa única, cada una con características que las diferencian entre ellas, en el sistema robótico se hará uso de la computadora de placa única denominada como Raspberry Pi, en su versión 3 modelo B+, que nos proporciona entre otras cosas un procesador de 4 núcleos 64-bits con una velocidad de 1.4GHz, 1 GB de memoria *RAM*, *Wireless Lan* (*wifi*, *IEEE* 802.11.b/g/n/ac) en doble banda (2,4 y 5 GHz), *Bluetooth* 4.2/BLE, un puerto Gigabit Ethernet, 40-pin *GPIO* (*General Purpose Input/Output*, Entrada/Salida de Propósito General), un puerto HDMI, 4 puertos *USB* 2.0, 1 puerto para cámara *CSI*, un puerto para *DSI display* (*touchscreen display*), 1 salida estéreo de 4 polos y puerto de video compuesto, 1 puerto para Micro SD que sirve como memoria principal de la Raspberry Pi.



Figura 36. Raspberry Pi 3 B+

Fuente: (Raspberry, s.f.)

Para poner en funcionamiento la Raspberry se decidió hacer uso del sistema operativo (SO) proporcionado por la misma, por lo que se procedió a entrar en la página oficial, al enlace de descargas y se seleccionó Raspbian, que es un sistema operativo Linux basado en Debian, y se eligió la imagen de Raspbian con escritorio.

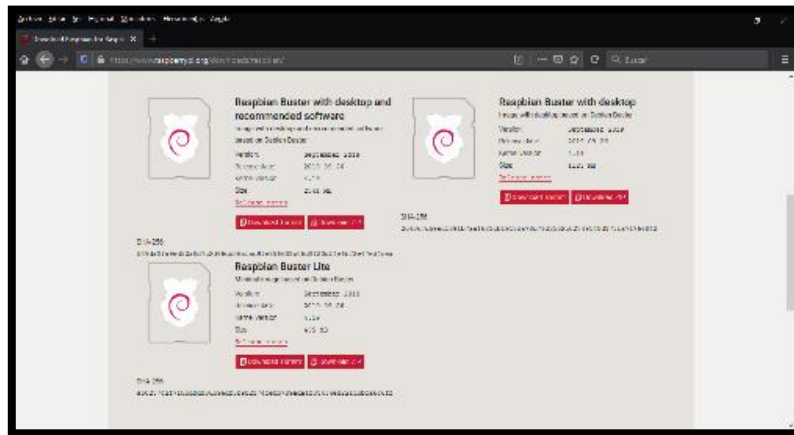


Figura 37. Sitio de descarga del SO Raspbian

Fuente: (Raspberry Pi, s.f.)

Una vez se descargó la imagen tal como sugiere la página de Raspberry se procedió a abrir el archivo usando el descompresor de archivos **7zip** (esto se realizó en un PC con Windows).

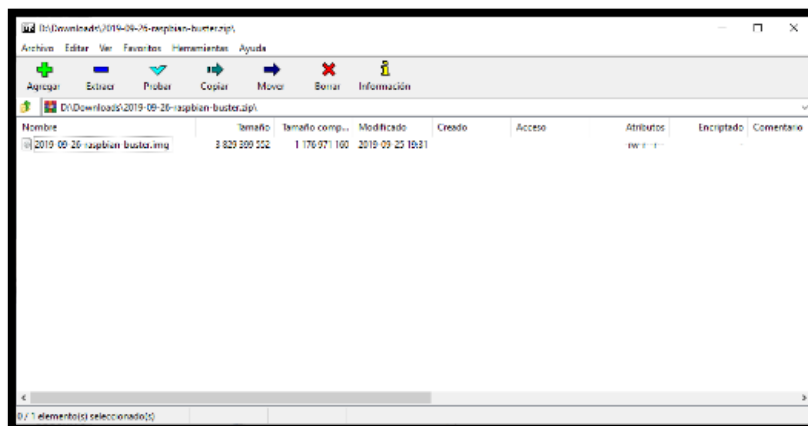


Figura 38. Programa 7zip

Fuente: [El Autor]

Se procedió luego a extraer el archivo img como se ve en la imagen.

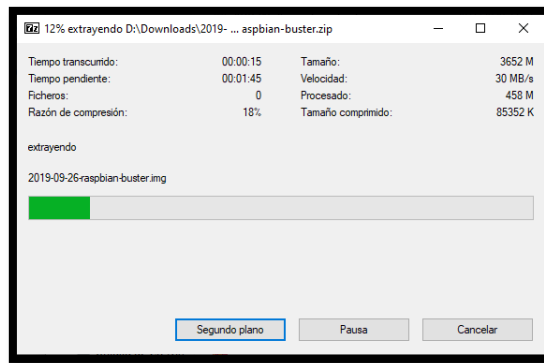


Figura 39. Extrayendo archivo img de Raspbian

Fuente: [El Autor]

Para instalar el archivo img de Raspbian en la memoria microSD se procedió a descargar el programa BalenaEtcher que es open Source desde su página oficial (la versión de Windows).

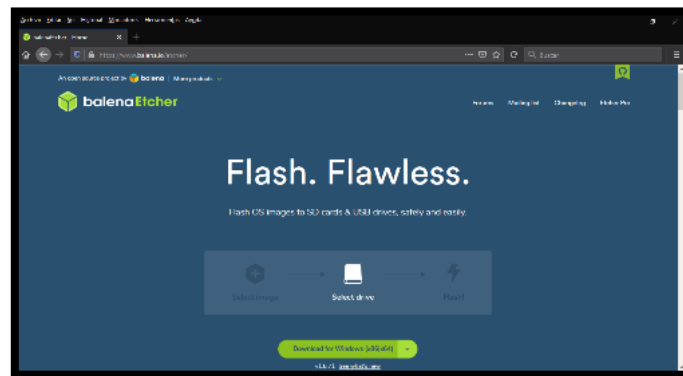
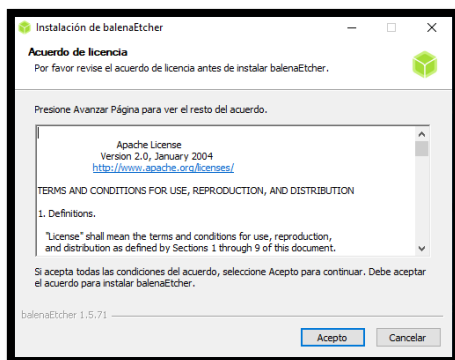


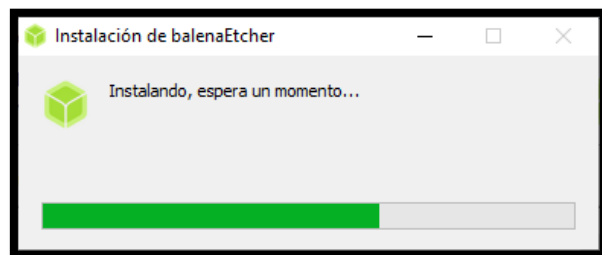
Figura 40. Página de BalenaEtcher

Fuente: (balenaEtcher, s.f.)

Una vez descargado el programa se continuó con la instalación del mismo, aceptando su acuerdo de licencia y se esperó a que termine.



(a)



(b)

Figura 41. A) Acuerdo de licencia del programa BalenaEtcher; b) Instalación del programa BalenaEtcher

Fuente: [El Autor]

Una vez instalado se procede a abrir el programa y se selecciona la imagen de Raspbian que se acabó de descargar y extraer anteriormente.

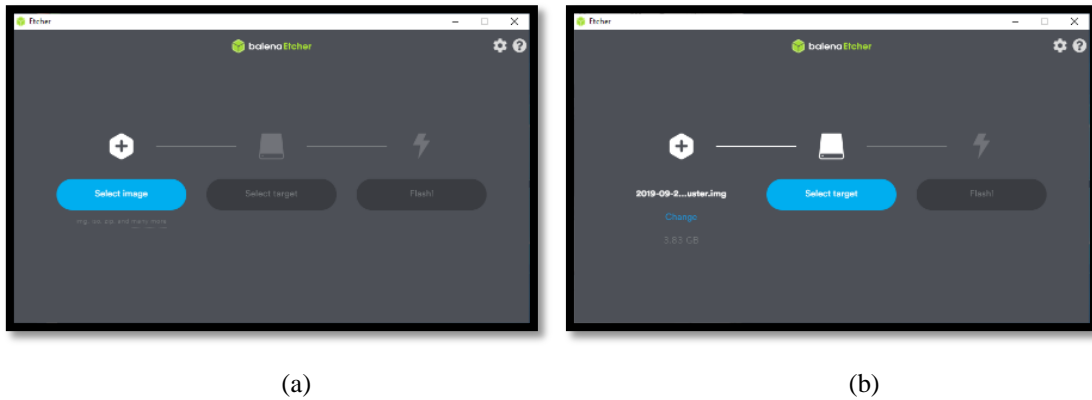


Figura 42. A) Pantalla principal del programa BalenaEtcher; b) Imagen de Raspbian seleccionada

Fuente: [El Autor]

Luego se debe seleccionar la unidad microSD, dicha microSD se seleccionó la Samsung Evo Plus de 64 GB.

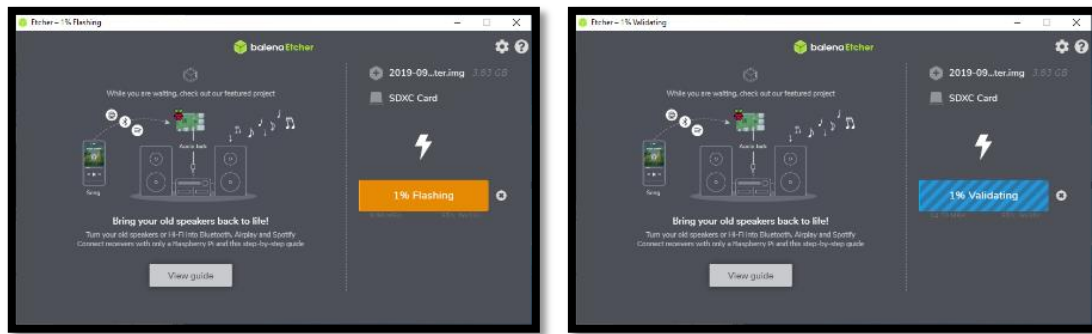


Figura 43. A) MicroSD Samsung Evo Plus 64 GB; b) Ubicación de microSD en el programa

BalenaEtcher

Fuente: [El Autor]

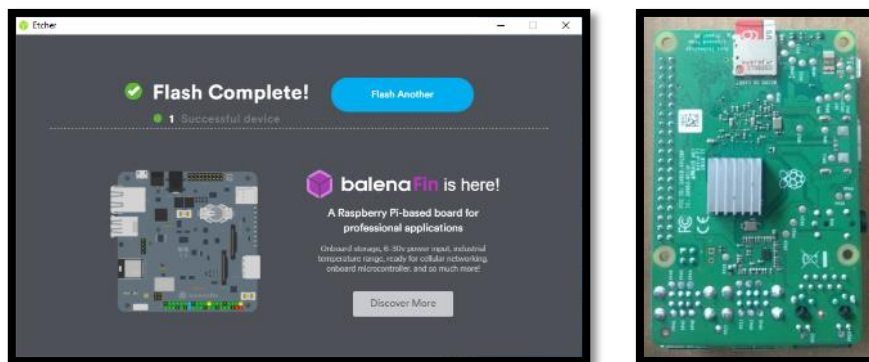
Y se procede a dar clic en el botón Flash, y se espera a que cree e instale y valide las particiones de *boot* y del sistema operativo Raspbian en la microSD.



(a) (b)  
 Figura 44. A) Instalando imagen de Raspbian en la microSD; b) Validando instalación

Fuente: [El Autor]

Una vez finalizada la instalación y la validación se procede a ubicar la microSD en la ranura específica para dichas tarjetas en la Raspberry Pi 3 B+.



(a) (b)

Figura 45. A) Instalación y validación finalizados; b) Ubicación de microSD en Raspberry Pi 3 B+

Fuente: [El Autor]

Luego de haber ubicado la tarjeta microSD en la Raspberry Pi 3 B+ se procedió a conectarla a un monitor y a la energía eléctrica y con la ayuda de un *mouse* y un teclado para poder realizar la configuración inicial.

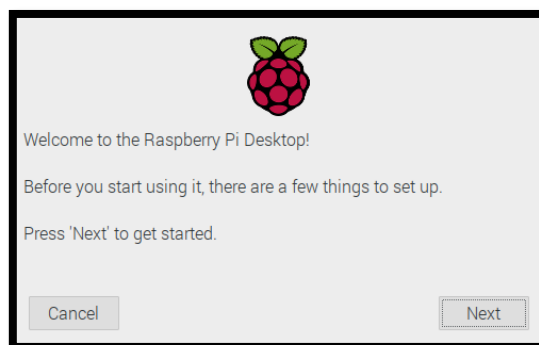


Figura 46. Ventana de configuración inicial de Raspbian

Fuente: [El Autor]

Al presionar el botón “*Next*” se nos mostró en la ventana las opciones para configurar el país en el que estamos, el idioma y la zona de tiempo, y se nos dio la elección de dejar el idioma del sistema operativo en inglés al igual que el idioma del teclado.

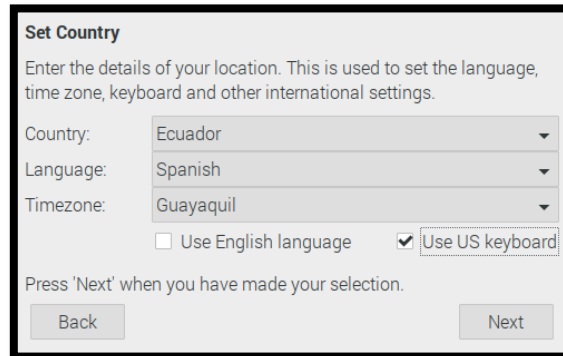


Figura 47. Configuración de país, idioma y zona de tiempo en configuración inicial de Raspbian

Fuente: [El Autor]

Luego presionamos el botón “*Next*” nos encontramos con la configuración para establecer una nueva contraseña para proteger nuestro sistema operativo.

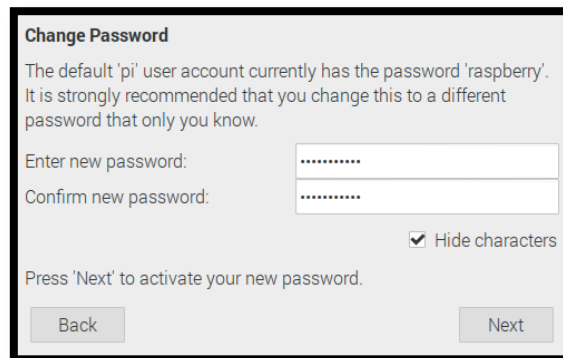


Figura 48. Establecimiento de nueva contraseña de Raspbian

Fuente: [El Autor]

Posteriormente presionamos el botón “*Next*” y podemos configurar dependiendo de si vemos que el escritorio que muestra nuestro sistema operativo Raspbian llena toda la pantalla o no se selecciona dicha opción.

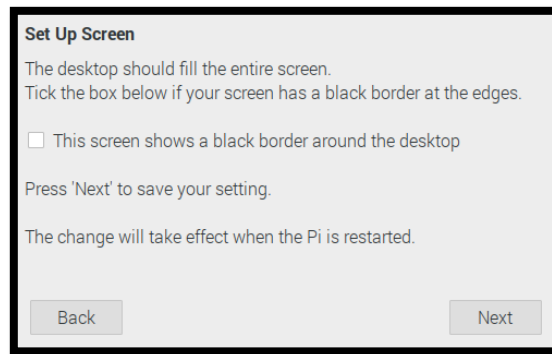


Figura 49. Llenar pantalla en la configuración inicial de Raspbian

Fuente: [El Autor]

Después de presionar el “*Next*” se nos mostró la ventana para la elección de una red WiFi la cual podemos seleccionar y continuar o presionar el botón “*Skip*” y saltarnos este paso.

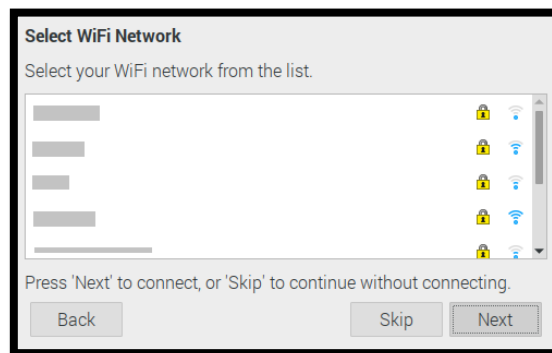


Figura 50. Elección de red WiFi en configuración inicial de Raspbian

Fuente: [El Autor]

Dependiendo de la ventana anterior se nos muestra una nueva ventana en la que pudimos digitar la contraseña asociada la red WiFi antes seleccionada, y también se nos da la opción nuevamente de saltarnos este paso con el botón “*Skip*”.

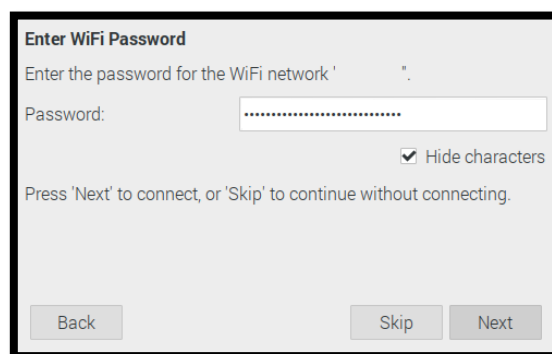


Figura 51. Introducción de contraseña de red WiFi en configuración inicial de Raspbian

Fuente: [El Autor]



Luego de digitar nuestra contraseña nos dirigirá a la ventana en la que se nos preguntó si queríamos actualizar el sistema operativo, se nos dio la opción de continuar o de saltar este paso.

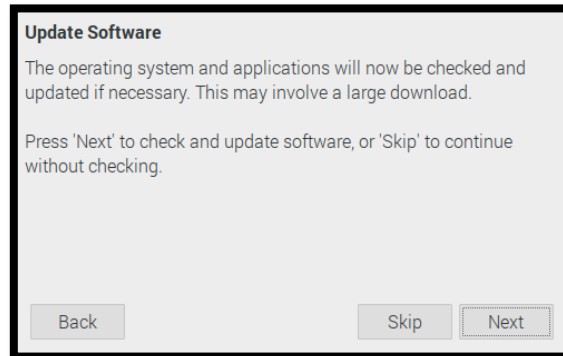


Figura 52. Actualización de software en configuración inicial de Raspbian

Fuente: [El Autor]

Por último, se nos preguntó si queríamos reiniciar el sistema operativo para aplicar algunos de los cambios hechos en la configuración inicial o hacerlo después.

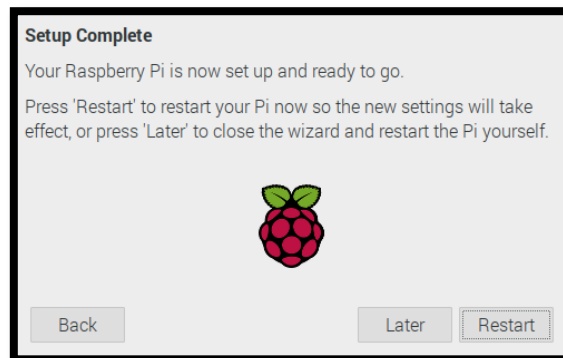


Figura 53. Reinicio de sistema en la configuración inicial de Raspbian

Fuente: [El Autor]

### 5.2.5.2. Arduino Mega

Para el control de actuadores (motores paso a paso) del sistema robótico automático se decidió el uso de la placa de desarrollo Arduino más concretamente la placa Arduino Mega que está basada en el microcontrolador AtMega, dicha placa cuenta con 54 digitales que pueden ser usadas como entradas o salidas para diferentes propósitos, 16 pines analógicos que son utilizadas comúnmente como entradas de datos de sensores, tiene una velocidad de reloj de 16 MHz.



Figura 54. Placa de desarrollo Arduino Mega

Fuente: (Arduino Mega, s.f.)

El Arduino Mega se lo usará como base para la ubicación de una placa que contara con los drivers DRV8825 de Pololu para el control de los motores paso a paso dicho driver cuenta con un límite de corriente ajustable, protecciones contra exceso temperatura y corriente, 6 tipos diferentes de resoluciones de micro pasos para el motor, el driver opera con un voltaje de entre 8,2V a 45V y hasta aproximadamente 1,5A sin usar un disipador o enfriamiento, con la cual podría funcionar hasta 2,2A sin problemas.

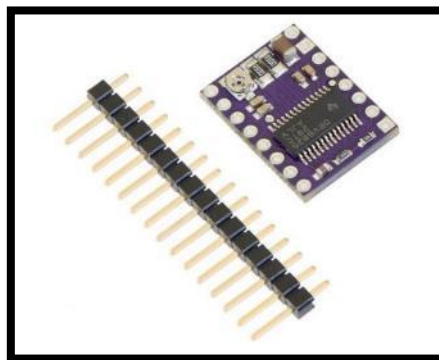


Figura 55. Driver DRV8825

Fuente: (DRV8825, s.f.)

Con ayuda del programa Proteus 8.1 se creó la placa para conectar los drivers DRV8825 a la placa de desarrollo Arduino Mega con mayor facilidad, usando la herramienta ISIS para seleccionar los elementos que se iban a utilizar y luego realizando la respectiva conexión, dado que entre los elementos que cuenta el programa no se encuentra ninguna pieza igual o parecida a los drivers DRV8825 se tuvo que utilizar en su lugar la pieza llamada “PIN” y usando la herramienta ARES se creó la placa PCB y se ubicaron los elementos “PIN” de modo que encajen con los pines de conexión de los drivers, una vez

hecho eso se creó un archivo de impresión para poder elaborar la placa PCB, luego de hacer las conexiones en ARES la placa quedó de la siguiente manera

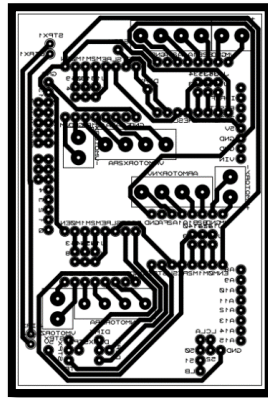


Figura 56. Circuito impreso de la placa de soporte de driver

Fuente: [El Autor]

Luego de trasladar el circuito impreso en una baquelita, usando ácido para extraer el exceso de cobre de la misma, se procedió a perforar la baquelita y a ubicar los elementos y soldarlos usando un caudín y estaño quedando de la siguiente manera

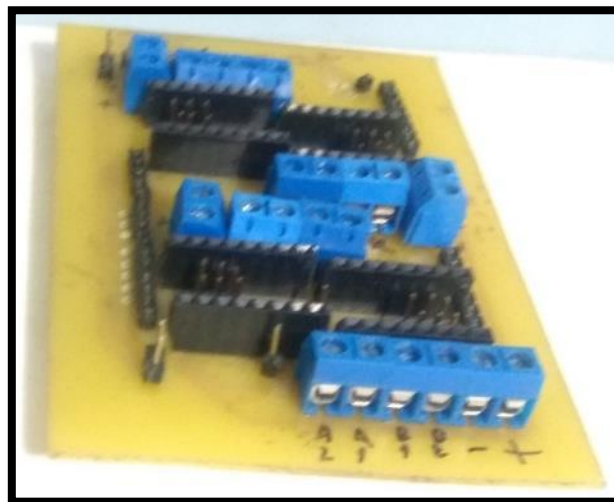


Figura Placa de soporte de drivers

Fuente: [El Autor]

### 5.2.5.3. Arduino Uno

Para el control de los motores paso a paso BYJ24 encargados de subir y bajar el sensor de humedad de tierra, el aspersor de agua y el sembrador de semillas y a su vez también controlar el sensado de los sensores de temperatura/humedad ambiental y humedad de tierra se utilizó la placa de desarrollo Arduino Uno basado en el microcontrolador AtMega

328 que tiene 14 pines digitales de los cuales 6 son pines PWM necesarios para el correcto funcionamiento de los motores y 6 pines analógicos dado que para los propósitos antes expuestos no se necesitan más pines, también cuenta con una velocidad de reloj de 16 MHz



Figura 57. Placa de desarrollo Arduino Uno  
Fuente: (Arduino Uno, s.f.)

El Arduino Uno se lo usará en conjunto con los drivers ULN2003 para controlar los motores paso a paso BYJ24

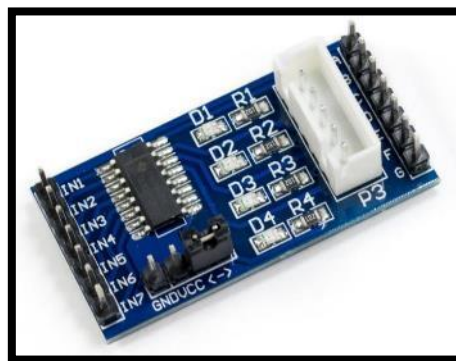


Figura 58. Modulo driver ULN2003  
Fuente: (Driver ULN2003, s.f.)

#### 5.2.5.4. Arduino Nano

Para el control de motores DC y motores vibradores usados para la construcción de los dosificadores de semillas se utilizó la placa de desarrollo Arduino Nano basado en el microcontrolador AtMega 328 que cuenta con 14 pines digitales 6 de ellos PWM y 6 pines analógicos, también tiene una velocidad de reloj de 16 MHz

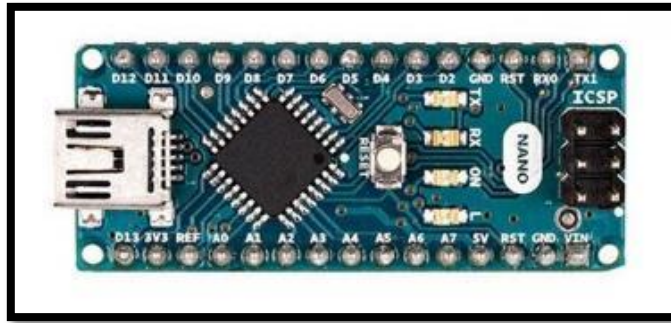


Figura 59. Placa de desarrollo Arduino Nano

Fuente: (Arduino Nano, s.f.)

El Arduino Nano se lo usa junto con los drivers L298N para controlar los motores DC

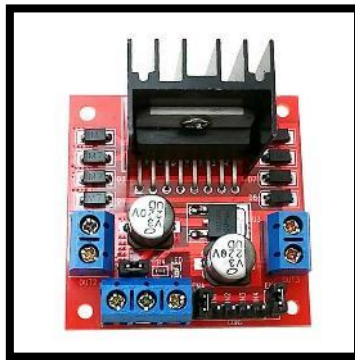


Figura 60. Modulo driver L298N

Fuente: (Driver L298N, s.f.)

## 5.2.6. Otros equipos

### 5.2.6.1. Equipo de riego

Dado que una de las tareas del sistema es encargarse de la siembra de semillas para obtener plántulas una de las necesidades para lograr esto es el riego por lo que se usó lo siguiente:

- **Electroválvula**

La electroválvula es un mecanismo electromagnético que nos permite controlar el paso de agua por una tubería al suministrarle energía eléctrica, la electroválvula que se usó en el sistema funciona con un voltaje de 120V en corriente alterna (AC); este elemento se lo usó para realizar el llenado de un balde para el posterior suministro de agua para riego.



Figura 61. Electroválvula

Fuente: [El Autor]

- **Sensor de nivel de agua**

El sensor de nivel de agua es un elemento electrónico que nos permite conocer la altura de un líquido dentro de un recipiente, el elemento que se usó en el sistema es un sensor de nivel de agua EP2508 tipo bolla, es decir, que tiene un módulo (bolla) que sube o baja dependiendo del nivel de agua y se comporta como interruptor permitiendo (bolla abajo) o negando (bolla arriba) el paso de energía eléctrica a través de los pines del sensor, permite cualquier nivel de voltaje hasta 100 V de corriente directa (DC) y una potencia máxima de 10W; el sensor se lo utilizara en conjunto con la electroválvula para el llenado del balde para riego según sea necesario.

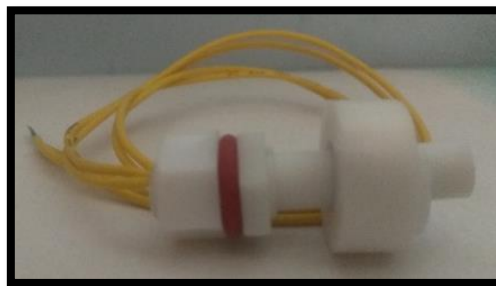


Figura 62. Sensor de nivel de agua EP2508

Fuente: [El Autor]

- **Bomba de agua**

La bomba de agua es un elemento que nos permite mover agua de un lugar a otro al suministrarle energía, la bomba de agua que se usó en el sistema bomba dc sumergible DC-1020 la cual funciona entre 5,5 a 12V de corriente directa (DC) y una potencia de 3W, y nos permite alcanzar una altura de entre 0,4 y 1,5 m para transportar el agua y realizar el riego necesario para el funcionamiento del sistema



Figura 63. Bomba de agua sumergible DC-1020

Fuente: [El Autor]

Los elementos usados para el riego en el sistema fueron controlados con la Raspberry Pi 3 B+ mediante el uso de los pines GPIO, con el limitante de que dichos pines funcionan hasta máximo 3,3V en entrada o salida, por lo que se creó una placa PCB para que los elementos funcionen con el voltaje correcto usando relés para permitir el uso de 12V DC usados por la bomba de agua y 120V AC usados por la electroválvula, transistores y resistencias usadas como interruptores para activar o desactivar los relés, por lo que la placa quedo de la siguiente manera:

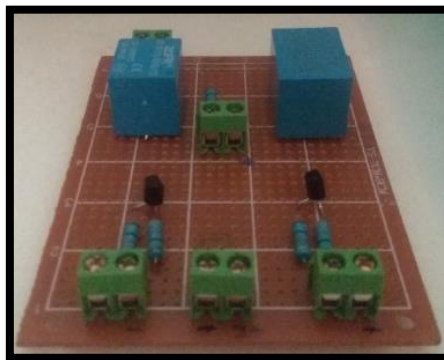


Figura 64. Placa PCB para equipos de riego

Fuente: [El Autor]

## 5.2.7. Software para control de actuadores y monitoreo de sensores

### 5.2.7.1. Control de actuadores

- **Motores paso a paso Pololu Nema 23**

Para el control de motores paso a paso Pololu Nema 23 se lo realiza usando un firmware Open-Source llamado GRBL que se encarga de transformar código G, que es un código utilizado en CNC, y luego enviar las debidas señales al driver DRV8825 y este realizara el movimiento de los motores en función del código utilizado, se descargó el firmware

GRBL desde su página en github procurando que sea el firmware para la placa de desarrollo Arduino Mega

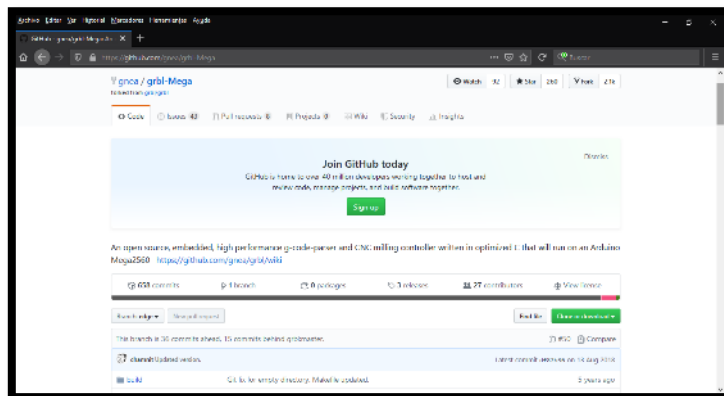


Figura 65. Página en github de firmware GRBL para Arduino Mega

Fuente: (grbl-Mega, s.f.)

Se utilizó el software Arduino IDE para instalar el firmware en la placa de desarrollo Arduino mega por lo que luego de descargar el firmware se procedió a instalarlo en las librerías de Arduino, esto se hace copiando la carpeta llamada “grbl” en el directorio de librerías de Arduino, luego de esto se abrió Arduino se dio clic en “Archivo” luego se ubicó en ejemplos y se buscó grbl y se dio clic en “grblUpload” para abrir el programa de grbl se procede a conectar la placa de desarrollo Arduino Mega y se procede a cargar el programa

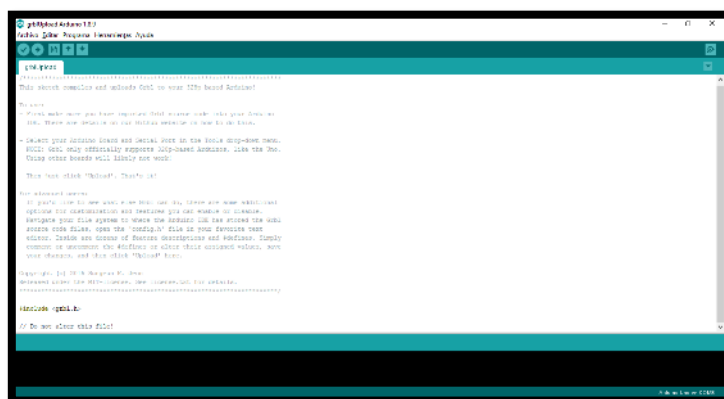


Figura 66. Programa en Arduino IDE para cargar el firmware GRBL

Fuente: [El Autor]

- **Motores paso a paso BYJ-24**

Para el control de los motores paso a paso BYJ-24 se desarrolló un programa en el entorno Arduino IDE, el cual envía las señales correspondientes desde la placa de desarrollo



Arduino Uno hacia los respectivos drivers ULN2003 y estos a su vez mueven los motores según sea conveniente.

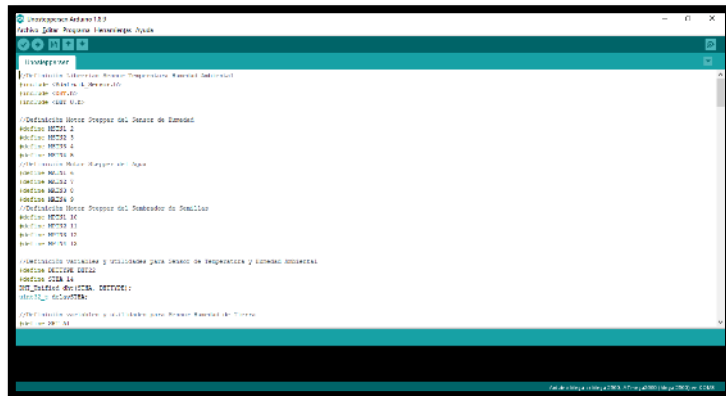


Figura 67. Programa para control de motores BYJ-24 en el entorno Arduino IDE

Fuente: [El Autor]

- **Motores DC y Vibradores**

Para el control de los motores DC y vibradores se desarrolló un programa en el entorno Arduino IDE, este programa se comunica con los drivers L298N que están conectados a los motores DC y estos a su vez giran según sea necesario en el dosificador de semillas; los motores vibradores en cambio están conectados directamente a los pines digitales y a tierra para que estos vibren o estén en reposo; el programa se desarrolló para la placa de desarrollo Arduino Nano.

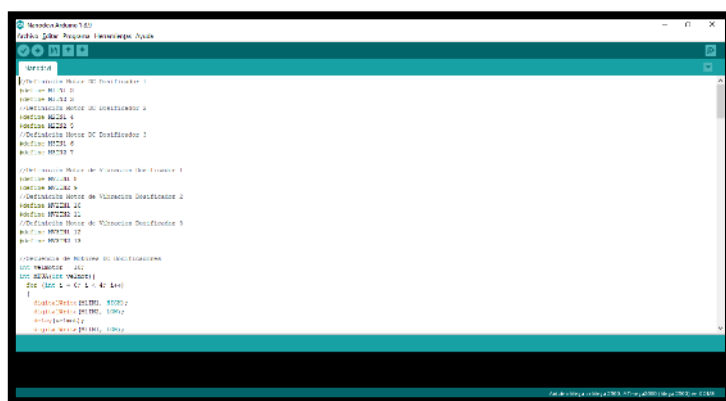


Figura 68. Programa de control de motores DC y vibradores en el entorno Arduino IDE

Fuente: [El Autor]

### 5.2.7.2. Software para monitoreo de sensores

- **Sensor de temperatura y humedad ambiental DHT-22**

Para el correcto funcionamiento de dicho sensor es necesaria la utilización de la librería llamada Adafruit, que puede ser descargada desde la página web de github de dicha librería

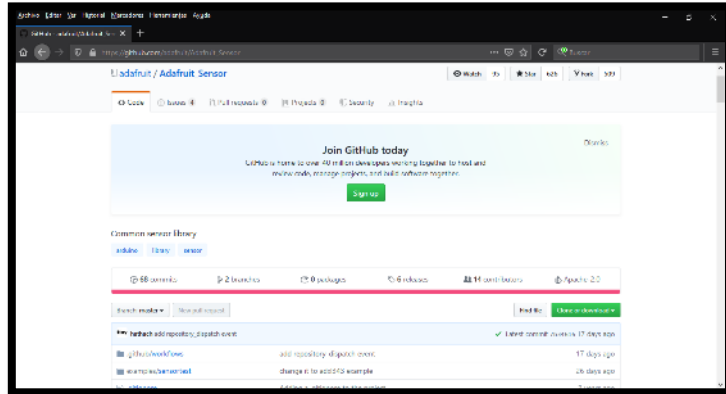


Figura 69. Página web GitHub de la librería Adafruit

Fuente: (Adafruit\_Sensor, s.f.)

Luego de descargar la librería de Adafruit en la página de web de se procedió a buscar la librería para el sensor en específico DHT, en la misma página de Adafruit en la sección del “readme” se encuentra el link a la librería necesaria

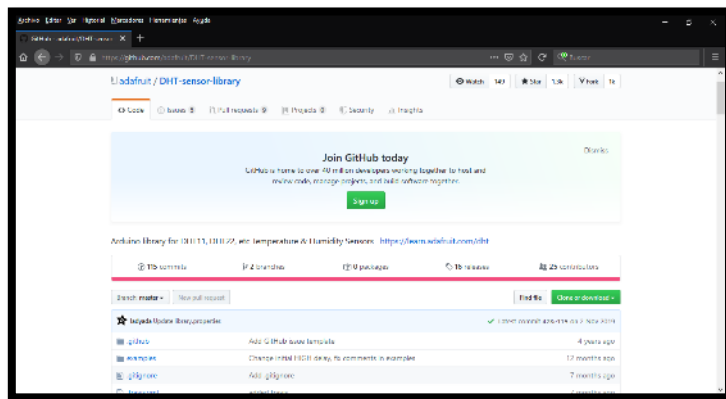


Figura 70. Página web en GitHub de la librería DHT- sensor-library

Fuente: (DHT-sensor-library, s.f.)

Una vez descargadas ambas librerías esta debieron ser instalada en el programa Arduino IDE en la carpeta correspondiente de librerías, una vez se instalaron se procedió a crear un programa para utilizar la librería se debió utilizar el comando “#include” para incluir la librería, tanto de “<Adafruit\_Sensor.h>” como las del sensor en específico “<DHT.h>” y <DHT\_U.h> para que el entorno funcione correctamente y pueda recoger los datos de temperatura y humedad ambiental que tome el sensor DHT-22

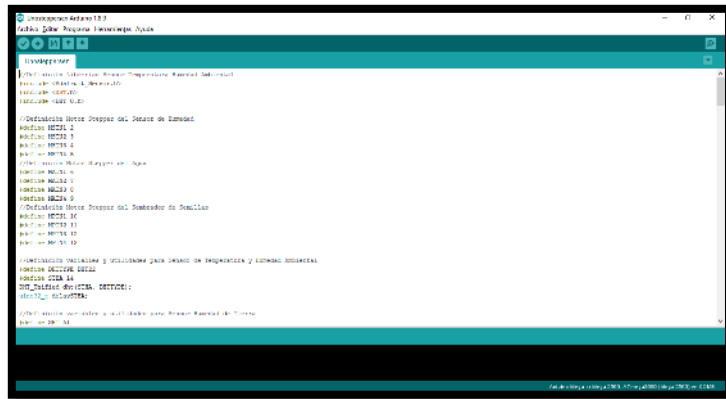


Figura 71. Programa para monitoreo de sensor DHT-22

Fuente: [El Autor]

- **Sensor de humedad de tierra YL-38**

Para el funcionamiento del sensor no es necesario instalarle ninguna librería extra al entorno Arduino IDE, basta con realizar una lectura del pin de entrada analógica en el que está conectado el pin de señal del sensor y transfórmalo a porcentaje

### 5.2.7.3. Software para automatización

#### ➤ CodeBlocks

CodeBlocks IDE es un entorno de desarrollo open-Source que nos permite la creación compilación y ejecución de programas de múltiples lenguajes de programación con sus respectivos compiladores, en el caso del sistema se usó el compilador GCC para realizar un programa en C++ que se encargó de la automatización del sistema, es decir, se encargara de sembrar semillas, regarlas, activar sensores para recoger datos y luego cargarlos a la base de datos, y también se encargó de la recepción de comandos cuando se utiliza la interfaz web (control manual). Para instalar CodeBlocks en nuestro sistema operativo Raspbian se abrió el terminal de Linux usando la combinación de teclas: “Ctrl+Alt+T”, luego se escribió en el terminal el comando siguiente:

```
pi@raspberrypi:~ $ sudo apt-get install codeblocks
```

Con lo cual se inició el proceso, el sistema operativo se encarga de buscar todos los archivos necesarios para instalar CodeBlocks IDE, tanto archivos del programa como librerías y programas adicionales necesarios, al terminar se nos pidió permiso para

descargar los archivos y ocupar espacio en la memoria de almacenamiento por lo que se teclea “s” y se procede a presionar “Enter”

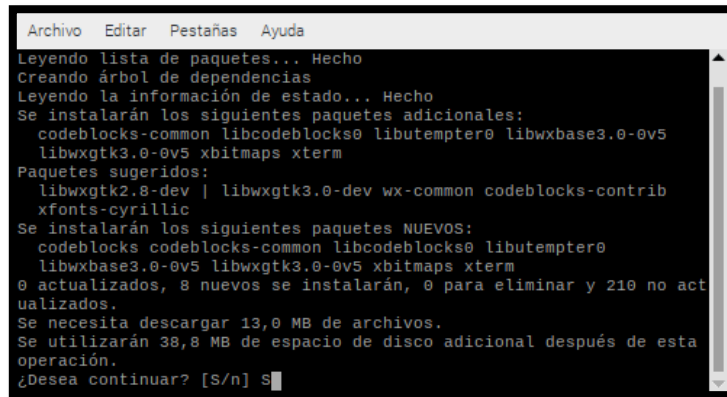


Figura 72. Ventana de instalación de CodeBlocks IDE, permiso de descarga y almacenamiento

Fuente: [El Autor]

Para el correcto funcionamiento de nuestro programa de automatización se necesitó la instalación de un compilador para C++, en el caso de Raspbian el paquete se llama “gcc”, normalmente este programa ya viene instalada en el sistema operativo descargado anteriormente, pero en caso de no se puede usar el siguiente comando:

```
pi@raspberrypi:~ $ sudo apt-get install gcc
```

En caso de que no se encuentre el compilador en el sistema se buscaran los archivos necesarios y se nos preguntará si se desea descargarlos, en caso de que si este instalado (este caso) se nos mostró que el compilador ya existe y en que versión se encuentra:

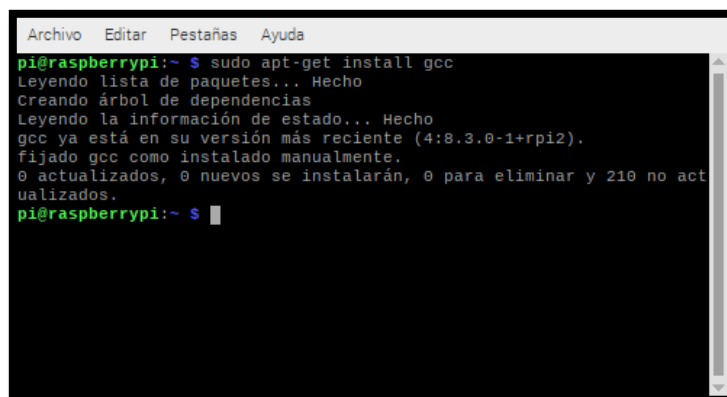


Figura 73. Compilador gcc ya instalado en Raspbian

Fuente: [El Autor]

Para que el programa de automatización funcione correctamente se hizo uso de las siguientes librerías:

### ➤ Librería Serial

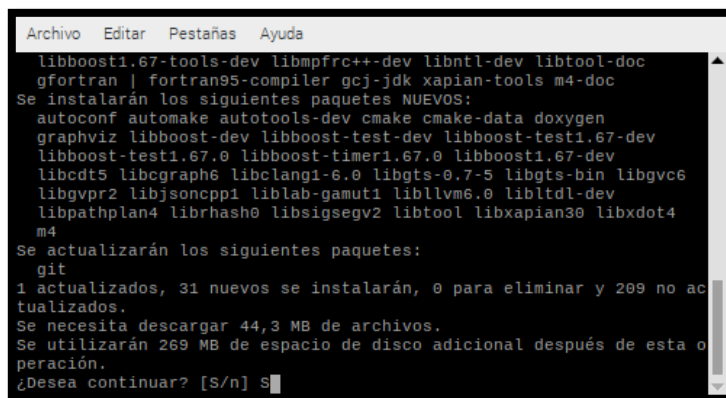
Esta librería nos ayudó a conectarnos (transmisión y recepción) mediante los puertos USB de la Raspberry Pi 3 B+ y los puertos USB-Serial con los que cuentan las plataformas de desarrollo Arduino para poder controlarlos mediante el programa de automatización creado en CodeBlocks escrito en lenguaje de programación C++. Para poder utilizarlo debimos instalar antes de ello programas y librerías que necesitábamos para luego instalar la librería en CodeBlocks, se lo realizo a través del terminal de Raspbian y usando el siguiente comando:

```
pi@raspberrypi:~ $ sudo apt-get install g++ git autoconf libboost-dev libboost-test-dev build-essential cmake doxygen graphviz libtool pkg-config
```

Figura 74. Comando de instalación de programas para descargar librería

Fuente: [El Autor]

Se espera a que reúna la información necesaria sobre las librerías y archivos necesarios, para empezar la instalación se nos pidió permiso con lo cual se tecleó “s” y se presionó “Enter”



```
libboost1.67-tools-dev libmpfr++-dev libntl-dev libtool-doc
gfortran | fortran95-compiler gcj-jdk xapian-tools m4-doc
Se instalarán los siguientes paquetes NUEVOS:
autoconf automake autotools-dev cmake cmake-data doxygen
graphviz libboost-dev libboost-test-dev libboost-test1.67-dev
libboost-test1.67.0 libboost-timer1.67.0 libboost1.67-dev
libcdt5 libcgraph6 libclang1-6.0 libgts-0.7-5 libgts-bin libgvc6
libgvpr2 libjsoncpp1 liblab-gamut1 libllvm6.0 libltdl-dev
libpathplan4 librtmp0 libsigsegv2 libtool libxapian30 libxdot4
m4
Se actualizarán los siguientes paquetes:
git
1 actualizados, 31 nuevos se instalarán, 0 para eliminar y 209 no ac
tualizados.
Se necesita descargar 44,3 MB de archivos.
Se utilizarán 269 MB de espacio de disco adicional después de esta o
peración.
¿Desea continuar? [S/n] s
```

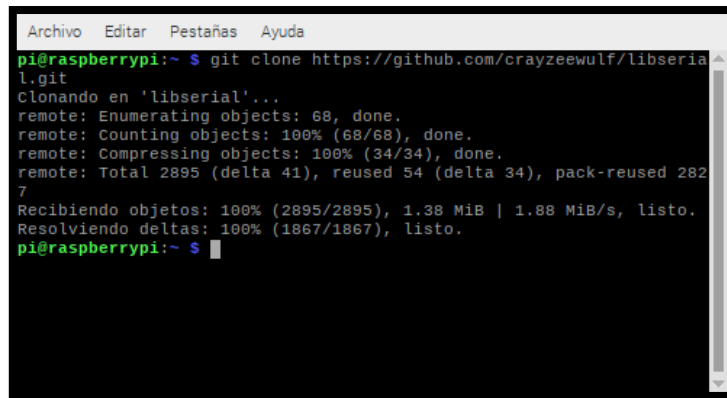
Figura 75. Instalación de programas y librerías necesarias

Fuente: [El Autor]

Una vez descargados los programas anteriores se procedió a descargar la librería LibSerial a través del siguiente comando:

```
pi@raspberrypi:~ $ git clone https://github.com/crayzeewulf/libserial.git
```

Al presionar “Enter” se empezó el proceso de descarga de la librería:



```
Archivo  Editar  Pestañas  Ayuda
pi@raspberrypi:~ $ git clone https://github.com/crayzeewulf/libserial
Clonando en 'libserial'...
remote: Enumerating objects: 68, done.
remote: Counting objects: 100% (68/68), done.
remote: Compressing objects: 100% (34/34), done.
remote: Total 2895 (delta 41), reused 54 (delta 34), pack-reused 2827
Recibiendo objetos: 100% (2895/2895), 1.38 MiB | 1.88 MiB/s, listo.
Resolviendo deltas: 100% (1867/1867), listo.
pi@raspberrypi:~ $
```

Figura 76. Descarga de librería LibSerial

Fuente: [El Autor]

Una vez descargado se entró en la carpeta de la librería usando el siguiente comando:

```
pi@raspberrypi:~ $ cd libserial/
pi@raspberrypi:~/libserial $
```

Para que cree los archivos necesarios para la instalación se usó el comando:

```
pi@raspberrypi:~/libserial $ make -f Makefile.dist
```

Luego para configurar la instalación se utilizó el comando:

```
pi@raspberrypi:~/libserial $ ./configure
```

Para pre instalar los archivos necesarios se usa el comando:

```
pi@raspberrypi:~/libserial $ make
```

Finalmente para instalar la librería LibSerial se usa el comando:

```
pi@raspberrypi:~/libserial $ sudo make install
```

Para que la librería en CodeBlocks IDE no nos ocasione problemas al compilar el programa de automatización se lo configuró entrando en CodeBlocks IDE luego se dio clic en el botón “Settings” luego “Compiler” luego se presionó la pestaña “Linker settings”:

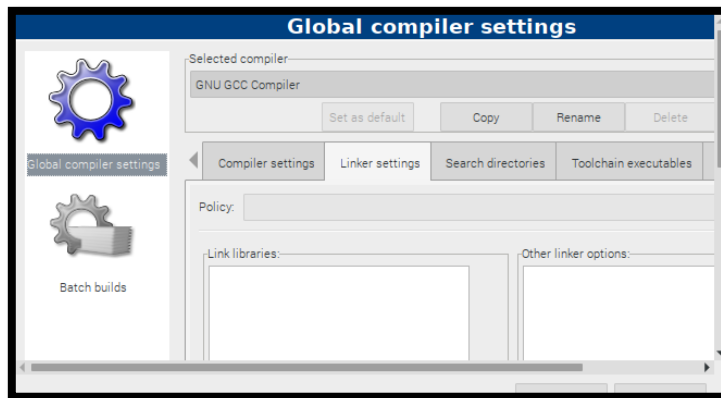


Figura 77. Configuración del compilador de CodeBlocks IDE

Fuente: [El Autor]

En esta ventana se dio clic en el botón “Add” que se encuentra en la parte inferior y nos abre la ventana en la que debimos dar clic en el icono con la carpeta para buscar los archivos “libserial.a” y “libserial.so” que normalmente se encuentran en la carpeta “/usr/local/lib” en el directorio de Raspbian:

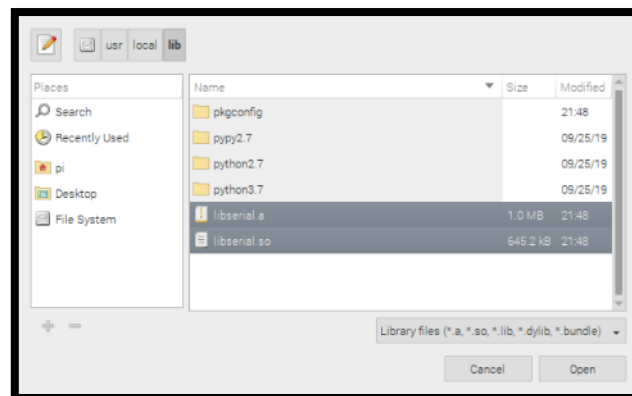


Figura 78. Ventana de búsqueda

Fuente: [El Autor]

Al presionar el botón “Open” se nos agregaron los archivos en la lista de enlaces de configuración, luego se da clic en “OK” que está en la parte inferior, con esto la librería ya puede ser utilizada sin ningún tipo de error.

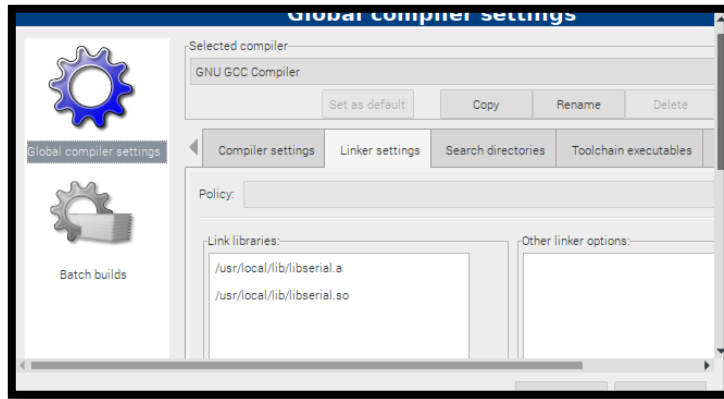


Figura 79. Ventana "Linker settings" con los archivos agregados

Fuente: [El Autor]

### ➤ Librería MySQL

La librería MySQL nos permitió conectar la base de datos (DB) y el programa de automatización, con ello podemos consultar, insertar, actualizar en otras cosas las tablas de la DB; para poder utilizarlo debimos usar el terminal del sistema operativo y teclear el siguiente comando:

```
pi@raspberrypi:~ $ sudo apt-get install libmysqlcppconn-dev libmariadb-dev libmariadb3 libmariadbclient-dev
```

Se nos pedirá confirmación de descarga e instalación por lo que escribimos la tecla "S" y presionamos "Enter"

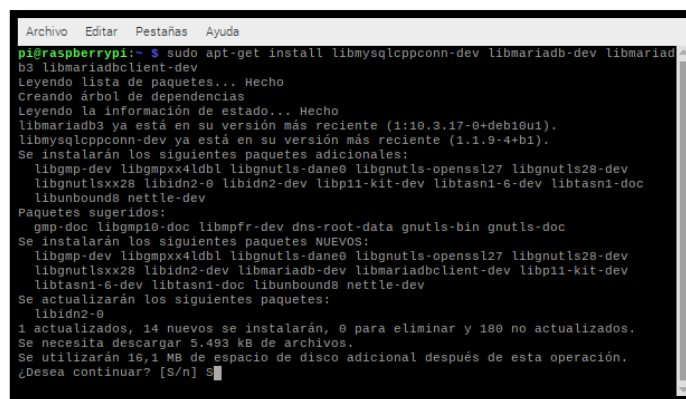


Figura 80. Instalación de librería MySQL Connector C++ y librerías necesarias

Fuente: [El Autor]

Una vez instalados los archivos necesarios debimos ubicar las librerías en CodeBlocks para que éste no nos genere errores, por lo que abrimos el programa, luego damos clic en "Settings" y luego en "Compiler" y en "Linker settings".



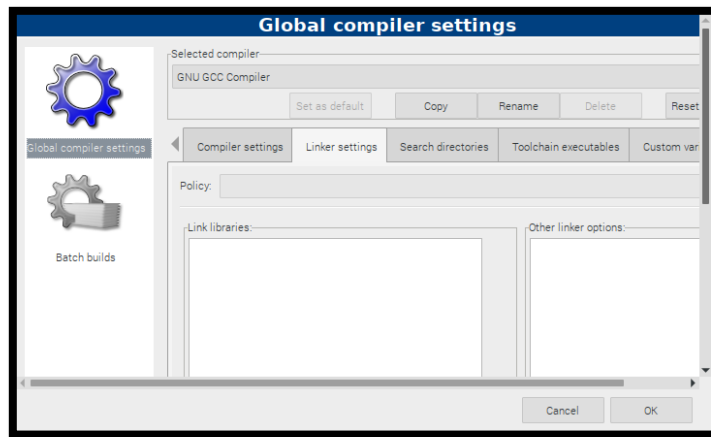


Figura 81. Configuraciones globales del compilador de CodeBlocks

Fuente: [El Autor]

Abierta la ventana nos desplazamos y damos clic en el botón “Add” y luego en botón con una carpeta para buscar las librerías que necesitamos, que generalmente se instalan en “/usr/lib/arm-linux-gnueabihf”, buscamos las librerías que debemos instalar que son: libdb.a, libdl.so, libmariadbclient.a, libmariadbclient.so, libmysqlcppcon.a, libmysqlcppcon.so, libmysqlcppcon-static.a, libpthread.so, libpthread\_nonshared.a, libz.a, libz.so.

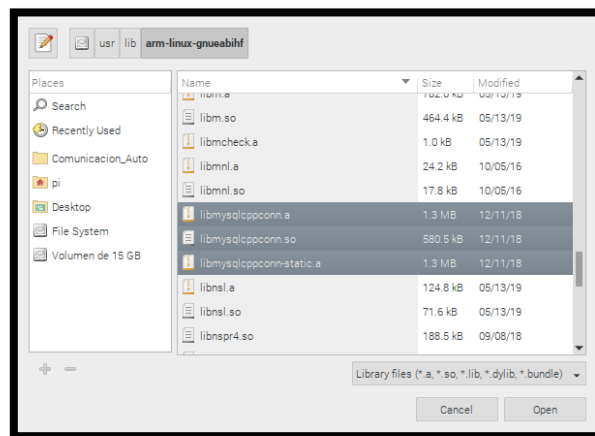


Figura 82. Librerías necesarias en CodeBlocks para MySQL Connector C++

Fuente: [El Autor]

Una vez seleccionadas todas las librerías presionamos el botón “Open” y luego el botón “Ok” para agregar las librerías a las configuraciones globales del compilador en CodeBlocks

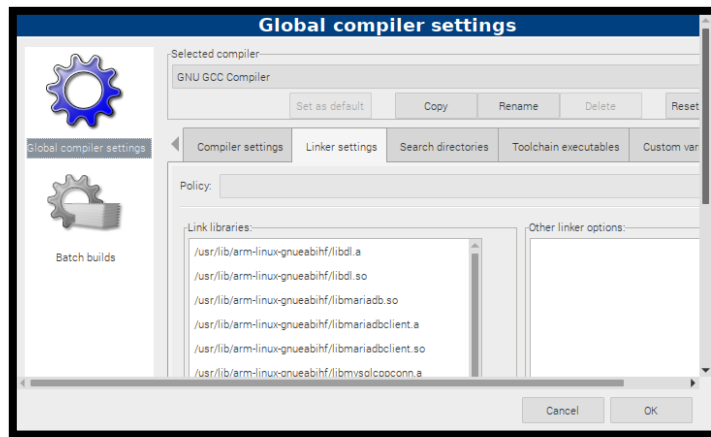


Figura 83. Configuraciones globales del compilador con las librerías agregadas

Fuente: [El Autor]

### ➤ Librería WiringPi

Esta librería nos dejó manejar los pines GPIO con los que cuenta la plataforma de desarrollo Raspberry Pi 3 B+, estos pines los utilizamos para controlar la electroválvula y la bomba de agua y monitorear el sensor de nivel de agua; esta librería normalmente viene instalada junto con el sistema operativo Raspbian por lo que podemos revisarlo usando el comando “gpio -v” en el terminal de Raspbian:

```

Archivo  Editar  Pestañas  Ayuda
pi@raspberrypi:~$ gpio -v
gpio version: 2.50
Copyright (c) 2012-2018 Gordon Henderson
This is free software with ABSOLUTELY NO WARRANTY.
For details type: gpio -warranty

Raspberry Pi Details:
  Type: Pi 3B+, Revision: 03, Memory: 1024MB, Maker:
  Sony
  * Device tree is enabled.
  *--> Raspberry Pi 3 Model B Plus Rev 1.3
  * This Raspberry Pi supports user-level GPIO acces
  s.
pi@raspberrypi:~$

```

Figura 84. Revisión de instalación de librería WiringPi

Fuente: [El Autor]

Si se encuentra instalado solo quedó configurarlo para usarlo en CodeBlocks IDE sin que nos de errores, se ejecutó el programa CodeBlocks IDE luego se dio clic en el menú “Setting” luego clic en “Compiler”, en la ventana de configuración del compilador se dio clic en la pestaña “Linker settings”:

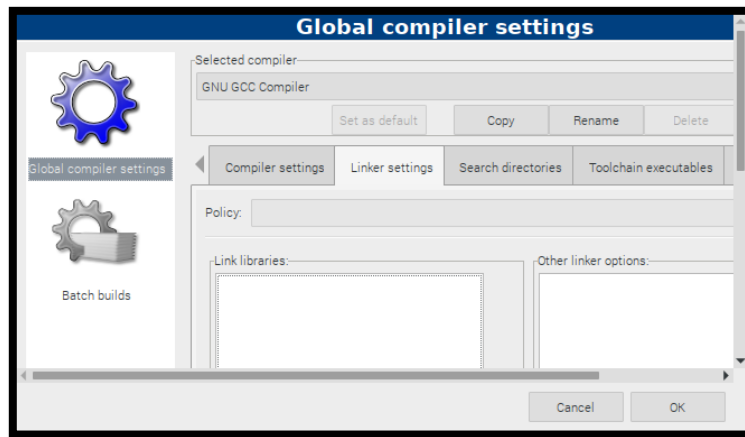


Figura 85. Ventana de configuración del compilador

Fuente: [El Autor]

Luego se presionó el botón “Add”, en la nueva ventana se dio clic en el icono con la carpeta para buscar los archivos “libwiringPi.so” y “libwiringPiDev.so” que normalmente se encuentran en “/usr/lib” o “/usr/local/lib” del directorio de Raspbian

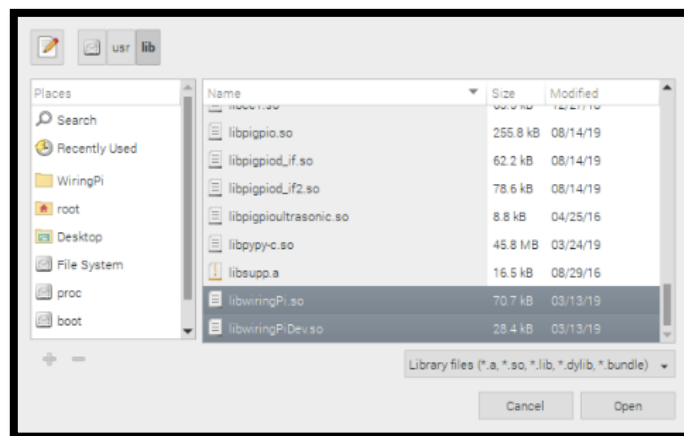


Figura 86. Ventana de selección de archivos

Fuente: [El Autor]

Luego se presionó el botón “Open” y luego “Ok”, con lo que la librería ya no nos generara ningún tipo de error al compilar el programa de automatización.

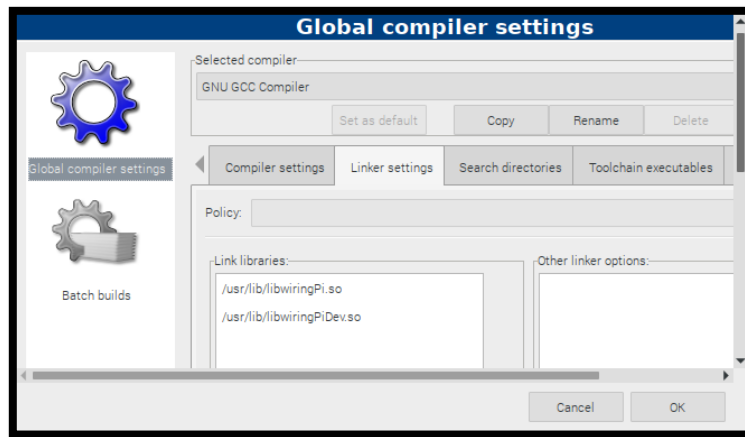


Figura 87. Configuración del compilador con los archivos agregados

Fuente: [El Autor]

En caso de que no esté instalado para poder usar la librería se debe seguir los pasos a continuación:

Se descargan los programas y librerías como se muestran las figuras: Figura 74 y Figura 75, luego se procedió a descargar la librería WiringPi usando el comando:

```
pi@raspberrypi:~ $ git clone https://github.com/WiringPi/WiringPi.git
```

Una vez se descargó se entró en la carpeta de la librería usando el comando:

```
pi@raspberrypi:~ $ cd WiringPi/  
pi@raspberrypi:~/WiringPi $
```

Al ser la primera vez que se descargó la librería se debió traer archivos extra usando el comando:

```
pi@raspberrypi:~/WiringPi $ git pull origin
```

Una vez hecho lo anterior se procedió a instalar la librería WiringPi usando el comando:

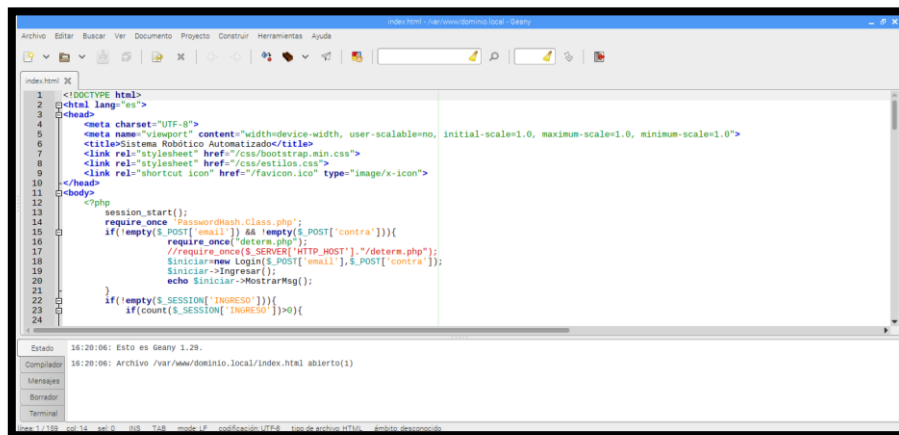
```
pi@raspberrypi:~/WiringPi $ ./build
```

Solo quedó configurar la librería en codeblocks IDE como se indicó anteriormente.

#### 5.2.7.4. Lenguaje de programación para desarrollo web

- **Html**

Es un lenguaje de marcado utilizado para la elaboración de páginas web, se usó este lenguaje para la creación de la interfaz web para control de actuadores (control manual) mediante botones y monitoreo de sensores mediante gráficas.



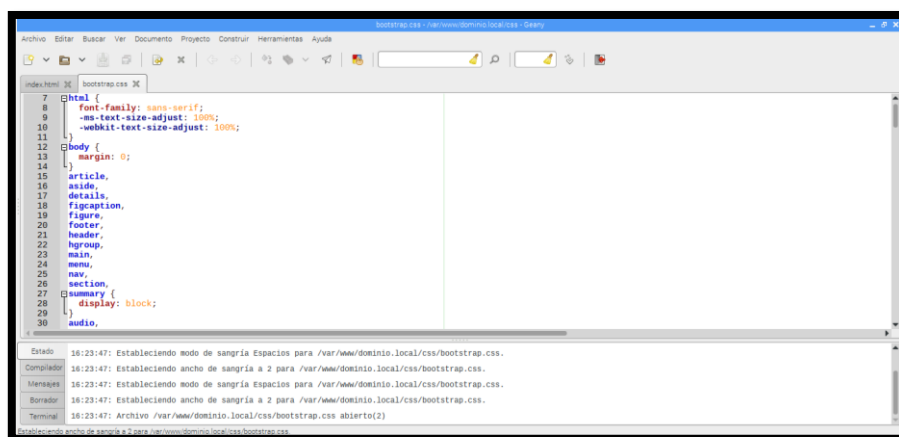
```
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
6 <title>Sistema Robótico Automatizado</title>
7 <link rel="stylesheet" href="/css/bootstrap.min.css">
8 <link rel="stylesheet" href="/css/estilos.css">
9 <link rel="shortcut icon" href="/favicon.ico" type="image/x-icon">
10 </head>
11 <body>
12 <?php
13 session_start();
14 require_once "passwordHash.class.php";
15 if(!empty($_POST['email']) && !empty($_POST['contra']))(
16     require_once "determ.php";
17     //require_once($_SERVER['HTTP_HOST']."/determ.php");
18     $iniciar=new Login($_POST['email'],$_POST['contra']);
19     $iniciar->Ingresar();
20     echo $iniciar->MostrarMsg();
21 }
22 if(!empty($_SESSION['INGRESO']))(
23     if(count($_SESSION['INGRESO'])>0){
24
```

Figura 88. Ejemplo: Página de inicio de la Interfaz Web

Fuente: [El Autor]

- **Bootstrap**

Es un conjunto de herramientas de código abierto que nos permite dar un mejor diseño a la página web con ayuda de sus plantillas, tipos de letra, formas, etc., se usó Bootstrap en la creación de la interfaz web en lo que fue los menús, botones, cuadros de texto, etc.



```
7 html {
8     font-family: sans-serif;
9     -ms-text-size-adjust: 100%;
10     -webkit-text-size-adjust: 100%;
11 }
12 body {
13     margin: 0;
14 }
15 article,
16 aside,
17 details,
18 figcaption,
19 figure,
20 footer,
21 header,
22 hgroup,
23 main,
24 menu,
25 nav,
26 section,
27 summary {
28     display: block;
29 }
30 audio,
```

Figura 89. Archivo bootstrap.css dentro del servidor web

Fuente: [El Autor]

Para instalarlo se debió descargar Bootstrap desde la página web de la misma, más concretamente los compilados de estilos CSS y los compilados de efectos, animaciones y acciones JS (JavaScript)

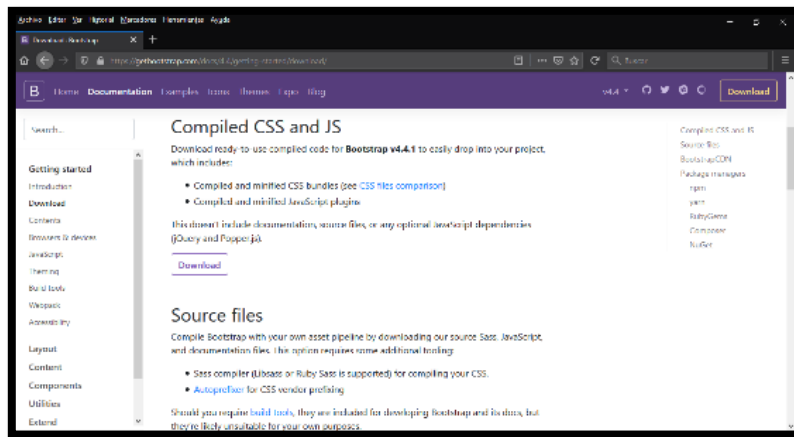


Figura 90. Página web de descarga de Bootstrap

Fuente: (BootStrap, s.f.)

Una vez descargado para poder utilizarlo en nuestra interfaz web las carpetas “css” y “js” se las copio dentro del directorio principal de servidor web.

- **PHP**

PHP es un lenguaje de programación usado para desarrollo web, se puede usar junto a html, entre las funciones que nos permite realizar están

- Conectar, consultar y actualizar la base de datos
- Control de ingreso por *login*
- Pre procesar datos

Si desea conocer como instalar PHP véase el subcapítulo 5.2.7.5 en la sección PHP.

### 5.2.7.5. Software para servidor web y almacenamiento de datos

#### ➤ LAMP

LAMP es un acrónimo para L “Linux”, A “Apache”, M “Mariadb”, P “PHP”, es un conjunto de sistemas y herramientas para la creación de un servidor web, aunque dichas herramientas se las puede usar de manera separada.

Antes de instalar LAMP se procedió a descargar un programa de firewall llamado “UFW” usando el comando:

```
pi@raspberrypi:~ $ sudo apt-get install ufw
```

Una vez instalado se procede a habilitar el firewall usando el comando:

```
pi@raspberrypi:~ $ sudo ufw enable
```

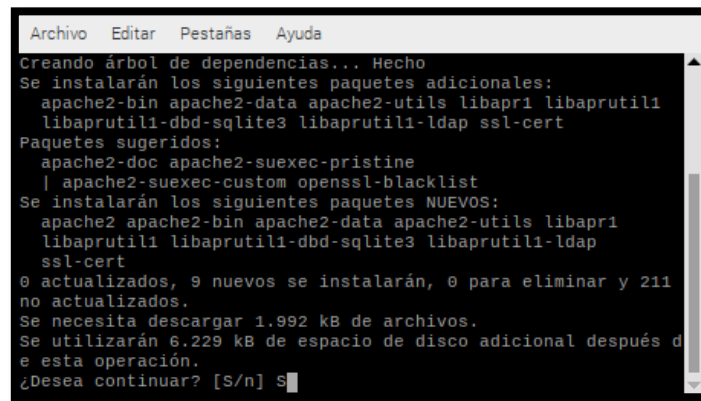
- **Apache**

Es un software de código abierto que sirve para la creación de servidores http o https, está disponible tanto en Linux como en Windows

Para instalar el servidor web se usó el siguiente comando en el terminal de Raspbian:

```
pi@raspberrypi:~ $ sudo apt-get install apache2
```

El sistema operativo reunirá la información necesaria de programas librerías y archivos necesarios para el funcionamiento de Apache y se nos pidió permiso para descargar e instalar los archivos, por lo que se tecleó “S” y luego se presionó “Enter”

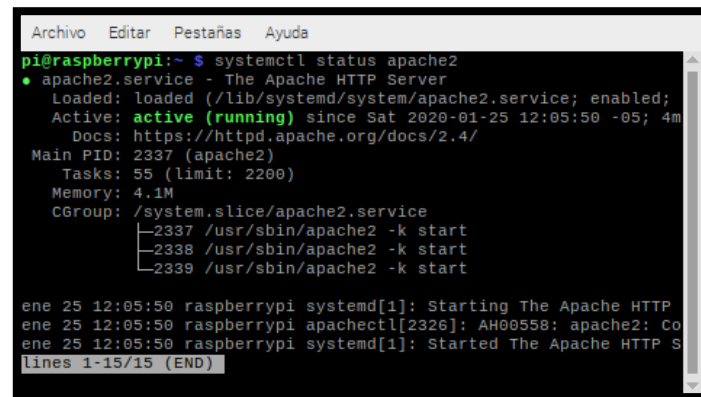


```
Archivo Editar Pestañas Ayuda
Creando árbol de dependencias... Hecho
Se instalarán los siguientes paquetes adicionales:
 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
 libaprutil1-dbd-sqlite3 libaprutil1-ldap ssl-cert
Paquetes sugeridos:
 apache2-doc apache2-suexec-pristine
 | apache2-suexec-custom openssl-blacklist
Se instalarán los siguientes paquetes NUEVOS:
 apache2 apache2-bin apache2-data apache2-utils libapr1
 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
 ssl-cert
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 211
no actualizados.
Se necesita descargar 1,992 kB de archivos.
Se utilizarán 6.229 kB de espacio de disco adicional después d
e esta operación.
¿Desea continuar? [S/n] S
```

Figura 91. Confirmación de instalación Apache

Fuente: [El Autor]

Una vez se instaló se comprobó el estado del servicio de Apache usando el comando “systemctl status apache2”:



```
Archivo Editar Pestañas Ayuda
pi@raspberrypi:~ $ systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled;
   Active: active (running) since Sat 2020-01-25 12:05:50 -05; 4m
   Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2337 (apache2)
   Tasks: 55 (limit: 2200)
   Memory: 4.1M
   CGroup: /system.slice/apache2.service
           └─2337 /usr/sbin/apache2 -k start
             └─2338 /usr/sbin/apache2 -k start
               └─2339 /usr/sbin/apache2 -k start

ene 25 12:05:50 raspberrypi systemd[1]: Starting The Apache HTTP
ene 25 12:05:50 raspberrypi apache2ctl[2326]: AH00558: apache2: Co
ene 25 12:05:50 raspberrypi systemd[1]: Started The Apache HTTP S
lines 1-15/15 (END)
```

Figura 92. Servicio de Apache encendido

Fuente: [El Autor]

Luego de instalar Apache se procedió a agregar permisos de conexión al firewall tanto para http y https usando los comandos:

```
pi@raspberrypi:~ $ sudo ufw allow www
```

```
pi@raspberrypi:~ $ sudo ufw allow https
```

Para realizar la comprobación de los permisos agregados se usó el comando “sudo ufw status”:



```
Archivo  Editar  Pestañas  Ayuda
pi@raspberrypi:~ $ sudo ufw status
Status: active

To Action From
-- --- --
80/tcp ALLOW Anywhere
443/tcp ALLOW Anywhere
80/tcp (v6) ALLOW Anywhere (v6)
443/tcp (v6) ALLOW Anywhere (v6)

pi@raspberrypi:~ $
```

Figura 93. Estado de firewall

Fuente: [El Autor]

Luego de instalar se debe deshabilitar la página por defecto que crea apache con el comando:

```
pi@raspberrypi:~ $ sudo a2dissite 000-default
```

Para crear nuestro sitio web, debemos hacer un archivo de configuración que leerá Apache, con lo que usamos el comando:

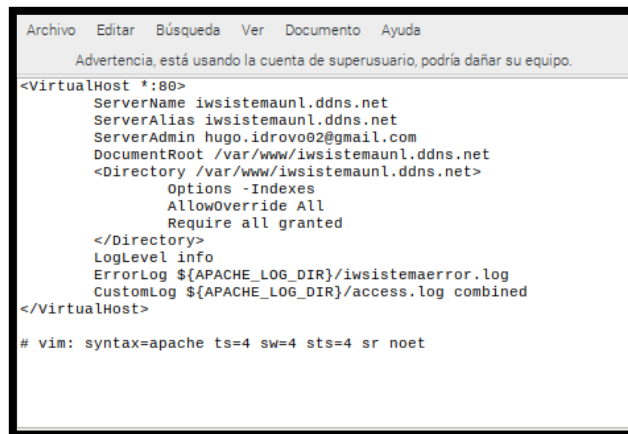
```
pi@raspberrypi:~ $ sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/iwsistemaunl.ddns.net.conf
```

Y abrimos el archivo creado usando el comando:

```
pi@raspberrypi:~ $ sudo mousepad /etc/apache2/sites-available/iwsistemaunl.ddns.net.conf
```

Para nuestra página web demos el archivo de la siguiente manera:





```
Archivo  Editor  Búsqueda  Ver  Documento  Ayuda
Advertencia, está usando la cuenta de superusuario, podría dañar su equipo.
<VirtualHost *:80>
  ServerName iwsistemaunl.ddns.net
  ServerAlias iwsistemaunl.ddns.net
  ServerAdmin hugo.idrovo02@gmail.com
  DocumentRoot /var/www/iwsistemaunl.ddns.net
  <Directory /var/www/iwsistemaunl.ddns.net>
    Options -Indexes
    AllowOverride All
    Require all granted
  </Directory>
  LogLevel info
  ErrorLog ${APACHE_LOG_DIR}/iwsistemaerror.log
  CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Figura 94. Archivo de configuración del sitio web en Apache

Fuente: [El Autor]

Por último para que Apache habilite nuestro sitio web y podamos abrir usamos el comando:

```
pi@raspberrypi:~ $ sudo a2ensite iwsistemaunl.ddns.net
```

Para que surta efecto la configuración debemos resetear Apache por lo que recurrimos al comando:

```
pi@raspberrypi:~ $ sudo systemctl restart apache2
```

En el segundo CD se encuentra la carpeta con los archivos usados para crear la interfaz web por lo que para copiarlos en nuestro sistema operativo usamos el comando:

```
pi@raspberrypi:~ $ sudo cp -r iwsistemaunl.ddns.net/ /var/www
```

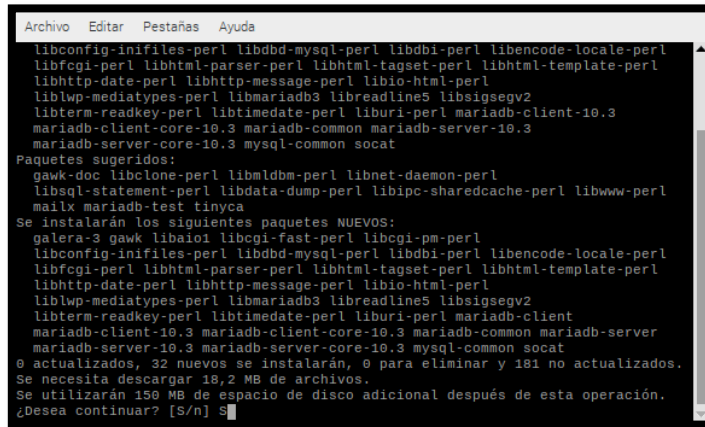
- **Servidor Mariadb**

Mariadb es un sistema open-Source que nos permite crear y gestionar bases de datos, está basado en MySQL, por lo que los comandos usados para consultar, actualizar, borrar, etc., la base de datos funcionan igualmente en Mariadb.

Para instalar el servidor MySQL Mariadb se usó el siguiente comando en el terminal de Raspbian:

```
pi@raspberrypi:~ $ sudo apt-get install mariadb-server mariadb-cl
ient
```

Se prepararán todos los archivos y programas necesarios para el funcionamiento de Mariadb Server, al acabar se nos pedirá permiso para descargar e instalar todos los archivos:

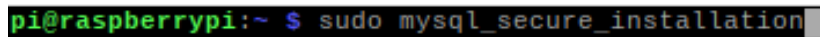


```
Archivo  Editar  Pestañas  Ayuda
libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libencode-locale-perl
libfcgi-perl libhtml-parser-perl libhtml-tagset-perl libhtml-template-perl
libhttp-date-perl libhttp-message-perl libio-html-perl
liblwp-mediatypes-perl libmariadb3 libreadline5 libsigsegv2
libterm-readkey-perl libtimedate-perl liburi-perl mariadb-client-10.3
mariadb-client-core-10.3 mariadb-common mariadb-server-10.3
mariadb-server-core-10.3 mysql-common socat
Paquetes sugeridos:
gawk-doc libclone-perl libmldbm-perl libnet-daemon-perl
libsql-statement-perl libdata-dump-perl libipc-sharedcache-perl libwww-perl
mailx mariadb-test tinyca
Se instalarán los siguientes paquetes NUEVOS:
galera-3 gawk libaio1 libcgi-fast-perl libcgi-pm-perl
libconfig-inifiles-perl libdbd-mysql-perl libdbi-perl libencode-locale-perl
libfcgi-perl libhtml-parser-perl libhtml-tagset-perl libhtml-template-perl
libhttp-date-perl libhttp-message-perl libio-html-perl
liblwp-mediatypes-perl libmariadb3 libreadline5 libsigsegv2
libterm-readkey-perl libtimedate-perl liburi-perl mariadb-client
mariadb-client-10.3 mariadb-client-core-10.3 mariadb-common mariadb-server
mariadb-server-10.3 mariadb-server-core-10.3 mysql-common socat
0 actualizados, 32 nuevos se instalarán, 0 para eliminar y 181 no actualizados.
Se necesita descargar 18,2 MB de archivos.
Se utilizarán 150 MB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
```

Figura 95. Permiso de descarga e instalación Mariadb

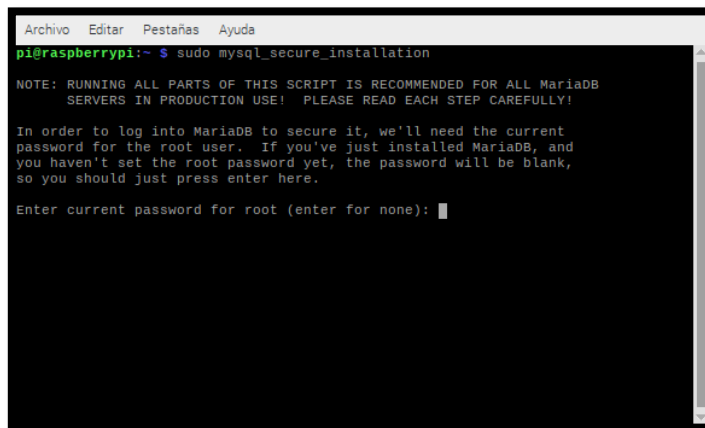
Fuente: [El Autor]

Una vez instalado debemos configurar el programa Mariadb usando el comando:



```
pi@raspberrypi:~ $ sudo mysql_secure_installation
```

Luego nos pedirá una contraseña de la base de datos



```
Archivo  Editar  Pestañas  Ayuda
pi@raspberrypi:~ $ sudo mysql_secure_installation
NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user.  If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
```

Figura 96. Introducción de contraseña actual

Fuente: [El Autor]

Dado a que se acaba de instalar y no se ha configurado una contraseña se procede a presionar la tecla “Enter”, entonces nos preguntará si queremos configurar una contraseña, a lo que respondemos tecleando “Y” y presionando “Enter”, e introducimos la nueva contraseña y luego una segunda vez

```
Archivo  Editar  Pestañas  Ayuda
pi@raspberrypi:~$ sudo mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user.  If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password: █
```

Figura 97. Introducción de contraseñas nuevas

Fuente: [El Autor]

Luego se nos preguntará si se desea remover los usuarios anónimos, que se crearon al instalar Mariadb para poder crear tablas de prueba, respondemos con la tecla “Y” y presionar “Enter”

```
Archivo  Editar  Pestañas  Ayuda
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them.  This is intended only for testing, and to make the installation
go a bit smoother.  You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] Y █
```

Figura 98. Eliminación de usuarios anónimos

Fuente: [El Autor]

Luego se nos preguntará si queremos deshabilitar el acceso remoto a la base de datos, respondemos con la tecla “Y” y presionar “Enter”

```
Archivo  Editar  Pestañas  Ayuda
Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them.  This is intended only for testing, and to make the installation
go a bit smoother.  You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] Y
... Success!

Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y
```

Figura 99. Deshabilitar acceso remoto

Fuente: [El Autor]

Después podemos remover las bases de datos de prueba y el acceso a ellas, respondemos con la tecla “Y” y presionar “Enter”

```
Archivo  Editar  Pestañas  Ayuda
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them.  This is intended only for testing, and to make the installation
go a bit smoother.  You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] Y
... Success!

Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access.  This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] Y
```

Figura 100. Eliminación de bases de datos de prueba

Fuente: [El Autor]

Finalmente se nos pregunta si se quiere recargar los privilegios a las tablas ahora, con ello toda la configuración que realizamos se guarde.

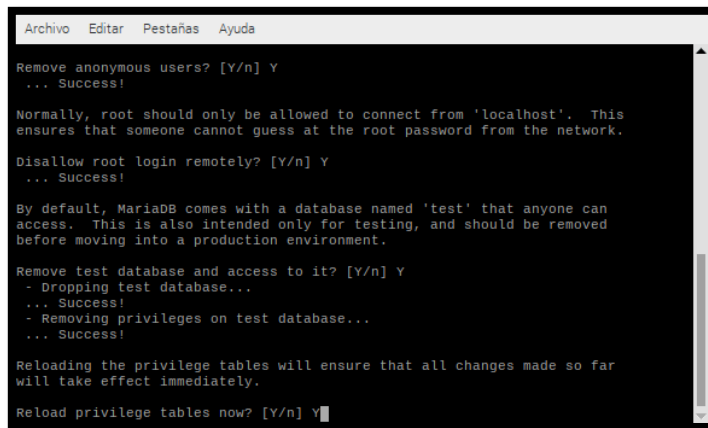


Figura 101. Recargar privilegios de tablas

Fuente: [El Autor]

Para una mejor y mas facil visualizacion y control de la base de datos se necesita una GUI mysql, por lo que se decidió el uso de PHPMYAdmin para este proposito; para descargarla tuvimos que usar el comando:

```
pi@raspberrypi:~ $ sudo apt-get install phpmyadmin
```

Se nos pide confirmación para descargar e instalar se responde con “S” y se teclea “Enter”, durante la instalación se nos mostrara una pantalla en la que nos pregunta que servidor web tenemos instalado (Apache en nuestro caso) para que PHPMYAdmin se configure con esos parámetros

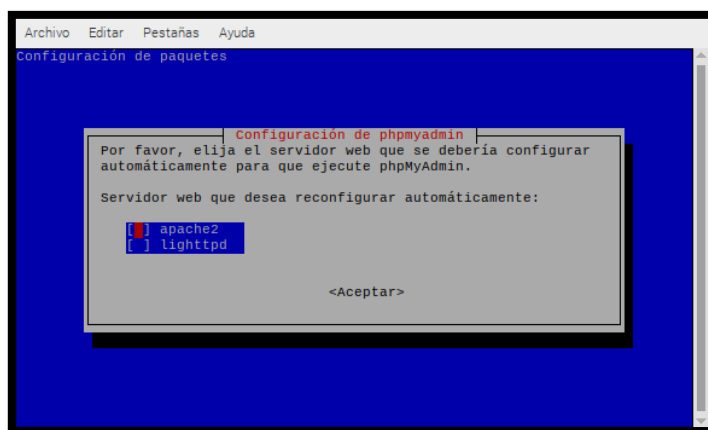


Figura 102. Configuración de PHPMYAdmin con respecto al servidor web

Fuente: [El Autor]

Se escoge y se teclea “Enter” para que continúe la instalación, en el trascurso de esta también se nos pedirá la contraseña que instalamos en MySQL para poder una base de

datos necesaria para el funcionamiento de PHPMyAdmin por lo que debemos proporcionársela

Una vez instalado todo podemos revisar si PHPMyAdmin funciona entrando a un navegador web y digitando la dirección “127.0.0.1/phpmyadmin” y se nos cargaría una página web como la de la Figura 104, en caso de que no cargue nada se debe dirigir al terminal del sistema operativo y usar el comando:

```
pi@raspberrypi:~ $ sudo ln -sf /usr/share/phpmyadmin/ /var/www/iwsistemaunl.ddns.net/
```

Con ello PHPMyAdmin se colocará en la raíz del servidor web y habilitará la página web para controlar la base de datos, en caso de que la raíz del servidor se encuentre en otra dirección solo se debe cambiar la última parte (“/var/www”) en consecuencia.

Antes de usar PHPMyAdmin debemos agregar las contraseñas y plugins en nuestra base de datos por lo que en el terminal usamos el comando:

```
pi@raspberrypi:~ $ sudo mysql -u root -p
```

Al dar “Enter” nos solicitará nuestra contraseña la cual escribimos y damos “Enter”, primero verificamos si las contraseñas y plugins están puestos usando el comando:

```
MariaDB [(none)]> SELECT user,authentication_string, plugin, host FROM mysql.user;
```

Obtendremos una tabla con dicha información

```
-----+
| user          | authentication_string          | plugin          | h
ost          |
+-----+-----+-----+-----+
-----+
| root          | *2470C0C06DEE42FD1618BB99005ADCA2EC9D1E19 | mysql_native_password | l
ocalhost    |
| root          |          |          | l
ocalhost    |
| root          |          |          | 1
27.0.0.1    |
| root          |          |          | :
:1          |
| phpmyadmin   |          |          | l
ocalhost    |
+-----+-----+-----+-----+
```

Figura 103. Tabla con usuario, cadena de autenticación, plugin y host en base de datos

Fuente: [El Autor]

Para ubicar las contraseñas junto con el plugin para el usuario “root” usamos los comandos (en donde dice “password” se cambia de acuerdo a la contraseña que se quiera):

```
MariaDB [(none)]> UPDATE mysql.user SET authentication_string = PASSWORD('password')
, plugin = 'mysql_native_password' WHERE user='root' AND host='localhost';
```

Para ubicar las contraseñas junto con el plugin para el usuario “phpmyadmin” usamos los comandos (en donde dice “password” se cambia de acuerdo a la contraseña que se quiera):

```
MariaDB [(none)]> UPDATE mysql.user SET authentication_string = PASSWORD('password')  
, plugin = 'mysql_native_password' WHERE user='phpmyadmin' AND host='localhost';
```

La base de datos creada para el proyecto cuenta con todas las tablas y datos necesarios para el correcto funcionamiento tanto de la interfaz web, como del programa de automatización, la base de datos se encuentra en el CD adjunto en formato “\*.sql”.

Para importar la base de datos a Mariadb se inició PHPMyAdmin desde un navegador usando la dirección de la interfaz web terminado en “/phpmyadmin” se nos muestra un proceso de login a lo que pones el usuario y contraseña de la base de datos Mariadb

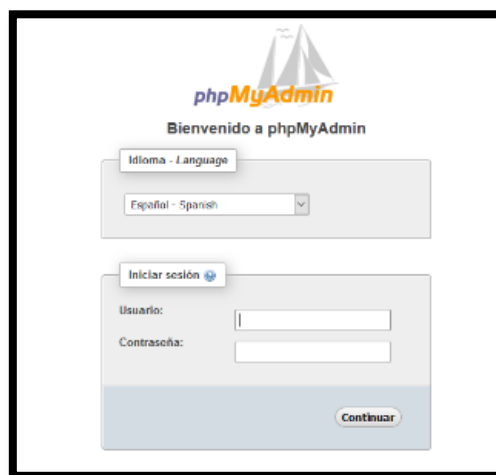


Figura 104. Interfaz web de PHPMyAdmin

Fuente: [El Autor]

Una vez se inició conexión con Mariadb y se tiene la interfaz de inicio de PHPMyAdmin, una vez aquí se debe dar clic en el botón “Import” para guardar la base de datos del sistema que se encuentra el archivo sql en nuestra base de datos Mariadb

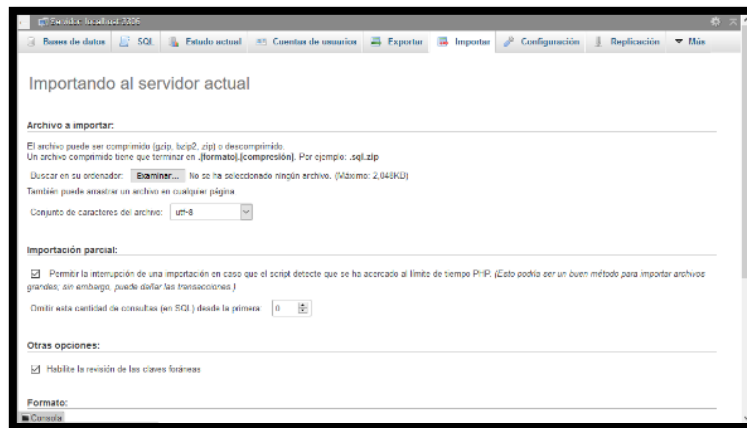


Figura 105. Interfaz para importar bases de datos PHPMyAdmin

Fuente: [El Autor]

Damos clic en el botón “Examinar” con lo cual se nos abrirá una ventana en la cual debemos buscar el archivo “Sist\_rob.sql” y dar clic en aceptar para abrir, una vez abierta se desliza la interfaz hasta el final en donde se presionó el botón “Continuar” para que empiece a crear la base de datos junto con las tablas y datos que contengan, si todo sale correctamente se nos mostrará en pantalla un mensaje.

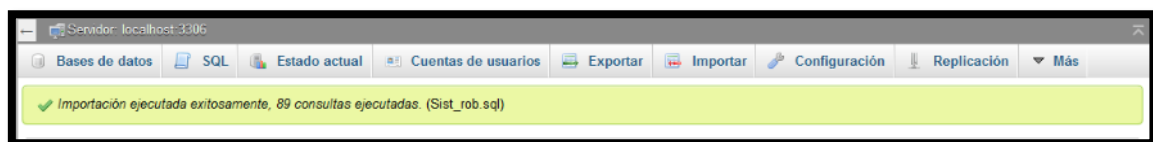


Figura 106. Importación ejecutada exitosamente

Fuente: [El Autor]

## ○ PHP

PHP es un lenguaje de programación para desarrollo web, PHP en LAMP nos permite en este caso como instalar herramientas en el servidor web para que pueda ejecutar los comandos usados en PHP (html), caso contrario la página web no cargaría y se generarían errores.

Para instalarlo debemos ejecutar el siguiente comando en la terminal de Raspbian:

```
pi@raspberrypi:~ $ sudo apt-get install php7.3 libapache2-mod-php7.3 php7.3-mysql php-common php7.3-cli php7.3-common php7.3-json php7.3-opcache php7.3-readline
```

Una vez instalado se debe digitar el siguiente comando:

```
pi@raspberrypi:~ $ sudo a2enmod php7.3
```



Dicho comando nos permite habilitar PHP en nuestro servidor web, ahora solo bastó con reiniciar el servicio Apache para que tome efecto la habilitación usando el comando:

```
pi@raspberrypi:~ $ sudo systemctl restart apache2
```

#### ➤ **Geany**

Geany es un editor de código y de texto, soporta muchos de lenguajes de programación, marcado y texto normal, este nos permite auto completar comandos y cerrar automáticamente etiquetas para elaborar códigos con mayor facilidad, también nos permite tener múltiples pestañas con diferentes archivos abiertos. Lo usamos para la programación de la página web en la creación de los códigos HTML, CSS y PHP.

Normalmente la distribución de Raspbian con software recomendado ya lleva instalado Geany de forma predeterminada, en caso de que no esté instalado se debió usar el siguiente comando en el terminal de Raspbian:

```
pi@raspberrypi:~ $ sudo apt-get install geany
```

Luego si se nos solicitara confirmación para descargar e instalar, se teclea “S” y se presiona "Enter".

### **5.2.7.6. Software DDNS**

#### ➤ **NO-IP**

No-Ip es un proveedor de servidores de dominio dinámicos, no -ip nos permite entre todos sus características el asociar la IP publica dinámica que nos asigna nuestro proveedor de servicio de internet a un dominio que se lo puede crear también en no-ip de manera gratuita, esto nos permite acceder a nuestra interfaz web desde cualquier otra red en la que nos encontremos para poder hacer la correcta monitorización de sensores, o consulta de información de plantas, al ser una IP publica dinámica cada cierto tiempo nuestro proveedor nos asigna una IP diferente por lo que adquirir el servicio de un proveedor de dominios normal cuando cambie la IP publica ya no podremos conectarnos a nuestra interfaz web con todo lo que eso conlleva.

Para poder usar No-Ip debemos dirigirnos a la página web oficial (<https://www.noip.com/>).



Figura 107. Página web de No-IP

Fuente: (No-IP, s.f.)

Una vez ahí damos clic en el botón “Sign Up”, en la nueva ventana creamos una cuenta nueva, en la que ponemos nuestro correo electrónico, una contraseña y ponemos el nombre del dominio que queremos crear, en la parte derecha nos da la opción de cambiar la última parte de nuestro dominio, luego en la parte inferior de la página damos clic en el botón “Free Sign up”, para terminar la creación.

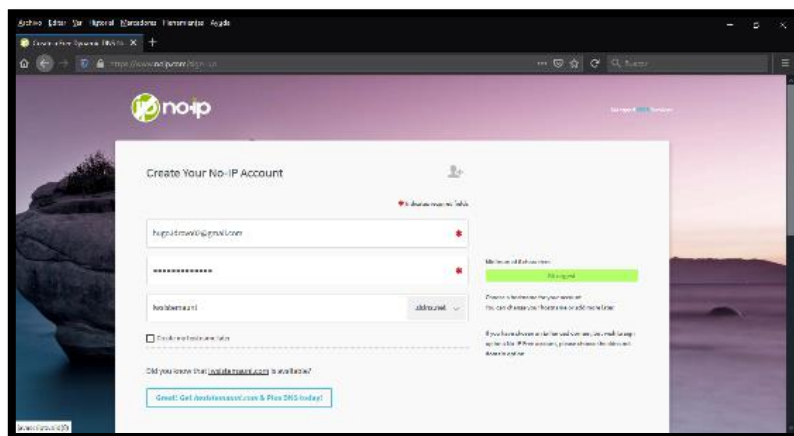
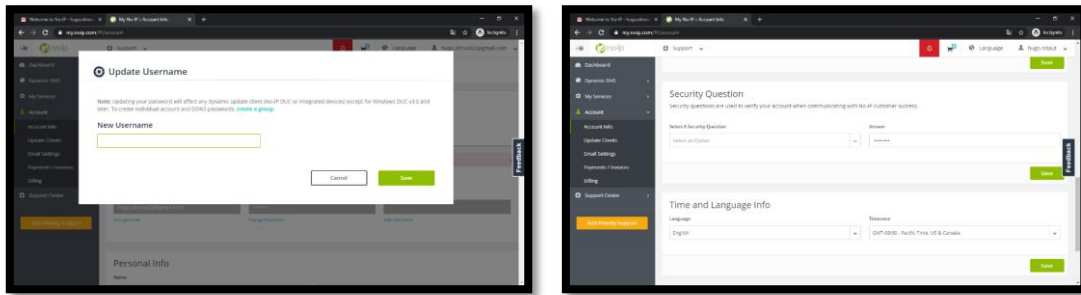


Figura 108. Creación de cuenta y dominio

Fuente: (No-IP, s.f.)

Se nos enviara a nuestro correo electrónico un email en el que debemos confirmar nuestra creación de cuenta, una vez confirmada se debe dar clic en “My Account” se nos abrirá un cuadro en el que nos dice que creamos un nombre de usuario y una pregunta de seguridad para completar nuestra cuenta en No-IP, por lo que las creamos, para crear el nombre de usuario damos clic en “Add username” y para la pregunta de seguridad nos

desplazamos en la página y seleccionamos la pregunta que queremos y damos en el botón “Save” para guardar



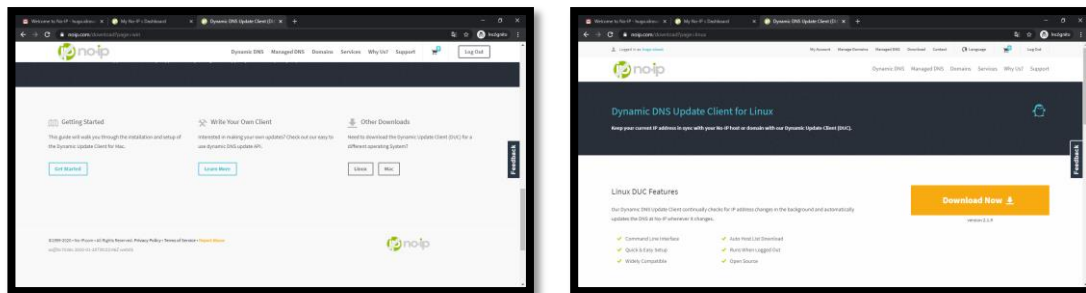
(a)

(b)

Figura 109. a) Actualización de nombre de usuario; b) Actualización de pregunta de seguridad

Fuente: (No-IP, s.f.)

Una vez listo debemos dirigirnos a la página web de inicio de No-IP y dar clic en la pestaña “Downloads” una vez ahí nos desplazamos por la página hasta donde dice “Other Download” y damos clic en Linux, dado a que el Sistema Operativo que estamos usando en nuestra Raspberry Pi, es basado en Linux Debian.



(a)

(b)

Figura 110. Sección de descargas de la página web de no IP

Fuente: (No-IP, s.f.)

Damos clic en el boton Download Now para empezar la descarga del programa y se espera a que se complete, despues el archivo se lo mueve a la carpeta del “/home/pi”, luego dado a que es un archivo comprimido se procede a darle clic derecho y luego al boton “Extraer aquí”

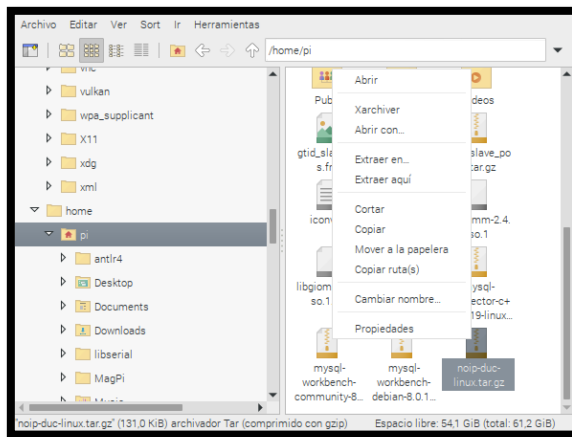


Figura 111. Extraer el archivo descargado de la pagina web de No Ip

Fuente: [El Autor]

Se abre el terminal del sistema operativo y se procede a abrir la ubicación del archivo extraído usando el comando:

```
pi@raspberrypi:~ $ cd noip-2.1.9-1/
```

Una vez en la carpeta se procede a compilar el programa usando el comando:

```
pi@raspberrypi:~/noip-2.1.9-1 $ sudo make
```

Una vez compilado se procede a instalarlo con el comando:

```
pi@raspberrypi:~/noip-2.1.9-1 $ sudo make install
```

Durante el proceso de instalación se procederá también a configurar el programa de No-Ip para que realice la asociación del nombre de dominio creado con anterioridad en la página web de No-Ip y nuestra IP pública por lo que nos pedirá que introduzcamos nuestro correo electrónico, nuestra contraseña, y se nos preguntara un intervalo para que el programa revise si nuestra IP publica cambio para asociar la nueva o mantener la actual en caso de que no, como sugerencia el programa pone 30 (minutos) pero podemos poner otro número.

```
Auto configuration for Linux client of no-ip.com.
Please enter the login/email string for no-ip.com
Please enter the password for user ' ' *****

Only one host [iwsistemaunl.ddns.net] is registered to this account.
It will be used.
Please enter an update interval:[30]
```

Figura 112. Configuración de programa de No-Ip

Fuente: [El Autor]

Para iniciar el programa se usa el comando:

```
pi@raspberrypi:~/noip-2.1.9-1 $ sudo /usr/local/bin/noip2
```

Y para revisar el estado del programa junto con la última IP pública, el hostname configurado, el tiempo de actualización, la interfaz de red usada, mediante el comando “sudo /usr/local/bin/noip2 -S”.

```
pi@raspberrypi:~/noip-2.1.9-1 $ sudo /usr/local/bin/noip2 -S
1 noip2 process active.

Process 11155, started as /home/pi/noip-2.1.9-1/noip2, (version 2.1.9)
Using configuration from /usr/local/etc/no-ip2.conf
Last IP Address set
Account
configured for:
    host iwsistemaunl.ddns.net
Executing username=                &pass=                &h[]=iwsistemaunl.ddns.net upon successful update.
Updating every 30 minutes via /dev/wlan0 with NAT enabled.
pi@raspberrypi:~/noip-2.1.9-1 $
```

Figura 113. Estado del programa de No-Ip

Fuente: [El Autor]

Para hacer que el programa de No-Ip se inicie automáticamente al encender la Raspberry Pi debemos crear un archivo en la carpeta “init.d” usando el comando:

```
pi@raspberrypi:~ $ sudo gedit /etc/init.d/noip2
```

Se abrirá un gestor de texto y en el ubicamos “sudo /usr/local/bin/noip2” quedando de la siguiente manera:

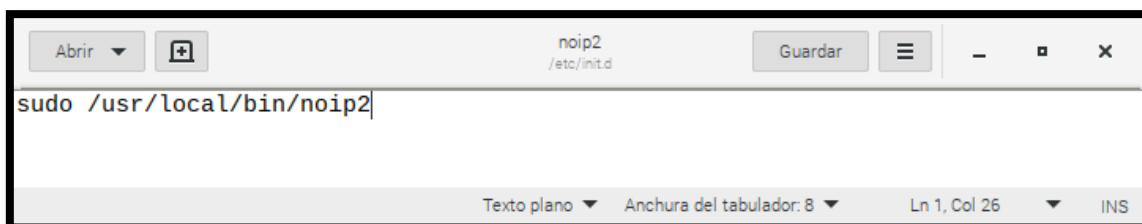


Figura 114. Archivo para iniciar automáticamente No-Ip

Fuente: [El Autor]

Luego le damos permisos de ejecución y de terminación, usando los comandos:

```
pi@raspberrypi:~ $ sudo chmod +x /etc/init.d/noip2
pi@raspberrypi:~ $ sudo update-rc.d noip2 defaults
```

## 6. RESULTADOS

### 6.1. Interfaz web

La interfaz web una vez que fue programada tanto a nivel lógico como a nivel diseño en la que se dividió en dos aspectos, uno de cara al usuario invitado, u otro de cara al usuario administrador del sistema.

#### 6.1.1. Usuario invitado

Dado que el sistema automatizado se encontraba en un servidor conectado a la red, cualquier persona con la *url* de la interfaz web podía acceder a ella, pero tenía limitadas las acciones con las que podía interactuar, a los usuarios invitados se les permitió revisar la información contenida en la interfaz web con respecto a las características edafoclimáticas investigadas en la presente tesis acerca de las plantas de siembra del cantón Loja y con respecto la visualización de los datos recogidos mediante los sensores elegidos para el sistema.

Al acceder a la *url* de la interfaz web nos encontramos con la página de inicio desde la cual nos podíamos dirigir hacia la información de las plantas o hacia los datos guardados de los sensores.

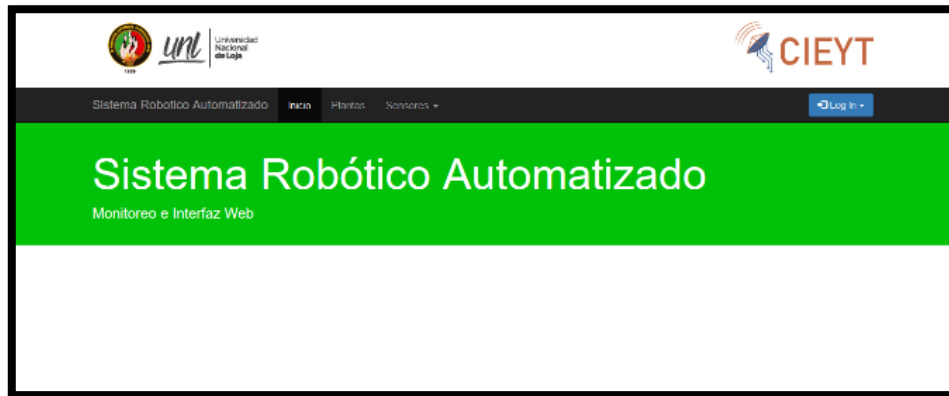


Figura 115. Interfaz web: Página de inicio usuario invitado

Fuente: [El Autor]

#### 6.1.1.1. Plantas

Al darle clic en el botón “Plantas” de la barra de menú nos redirigió a la página donde se encuentra guardada la información de las plantas.

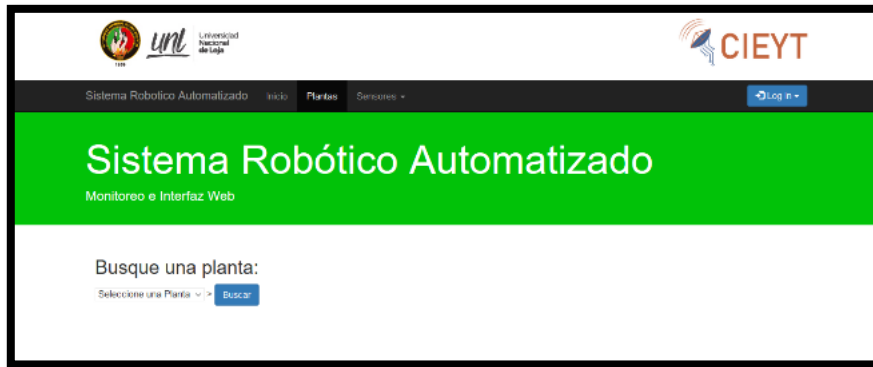


Figura 116. Interfaz web: Búsqueda de planta

Fuente: [El Autor]

En esta interfaz al presionar sobre el cuadro de selección de plantas se nos despliega una lista con todas las plantas que habían sido guardadas en la base de datos.

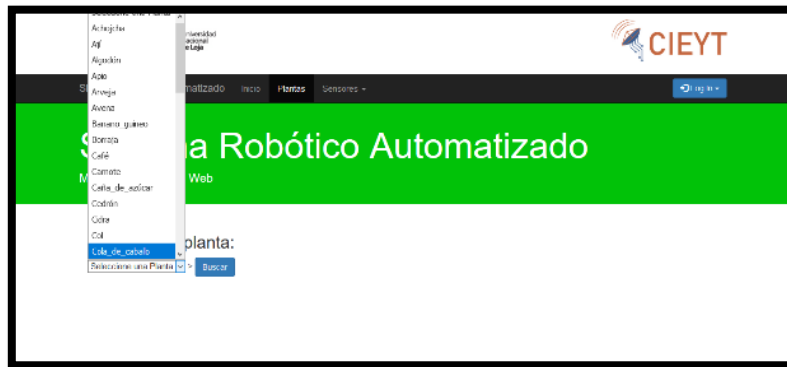


Figura 117. Interfaz web: Selección de una planta

Fuente: [El Autor]

Al seleccionar una planta de la lista y presionar el botón “Buscar” nos aparece un cuadro con toda la información recogida en la investigación acerca de la planta

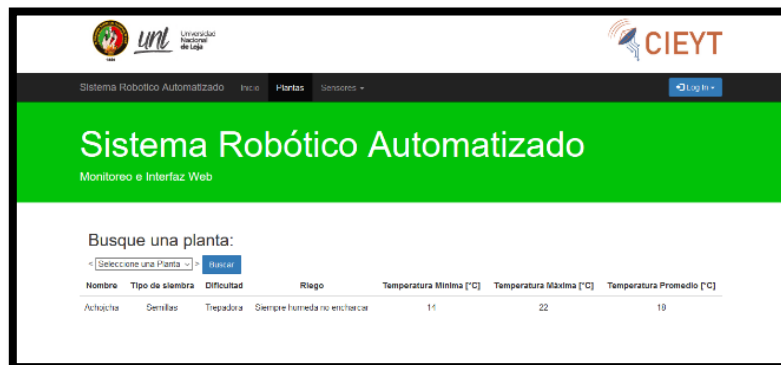


Figura 118. Interfaz web: Información de la planta

Fuente: [El Autor]

### 6.1.1.2. Sensores

Al dar clic en el botón “Sensores” se nos desplegó una lista de opciones para ingresar en la interfaz web específica para cada sensor y visualizar los datos guardados del mismo.

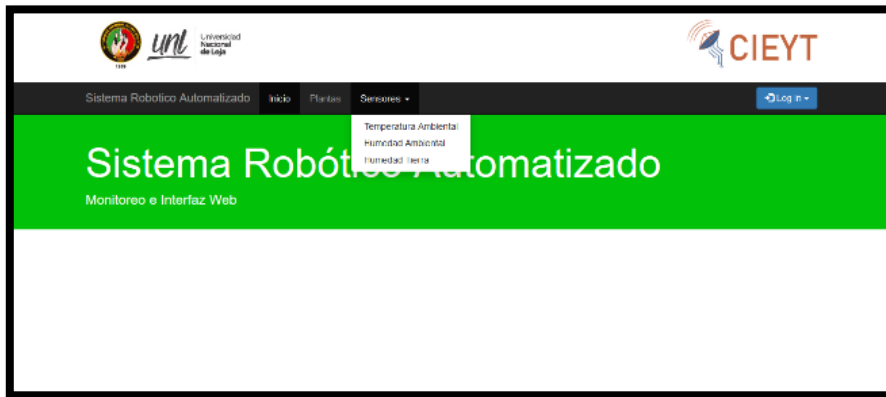


Figura 119. Interfaz web: Menú desplegable de Sensores

Fuente: [El Autor]

Cada interfaz web de sensores cuenta con dos selectores de “Fecha Y Hora” y un botón para realizar la búsqueda de los datos guardados en la base de datos en referencia al rango de fecha-hora elegido.

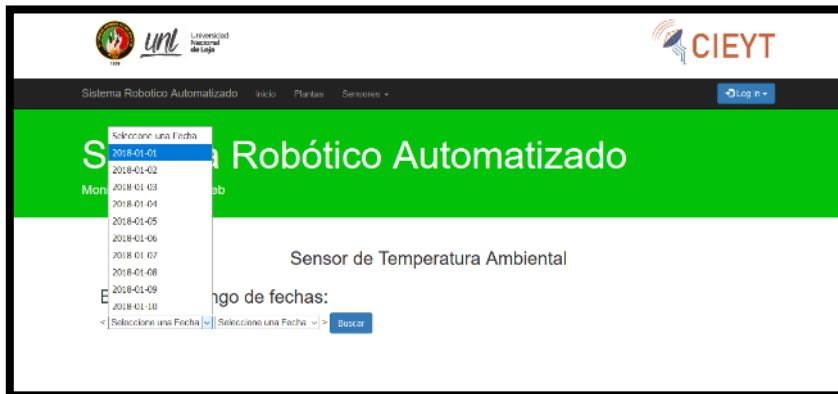


Figura 120. Interfaz web: Selección de intervalo de tiempo

Fuente: [El Autor]

Al elegir un rango de fecha-hora y presionar el botón buscar se despliega una gráfica del comportamiento de los datos recogidos por el sensor para una mejor visualización de los mismos.



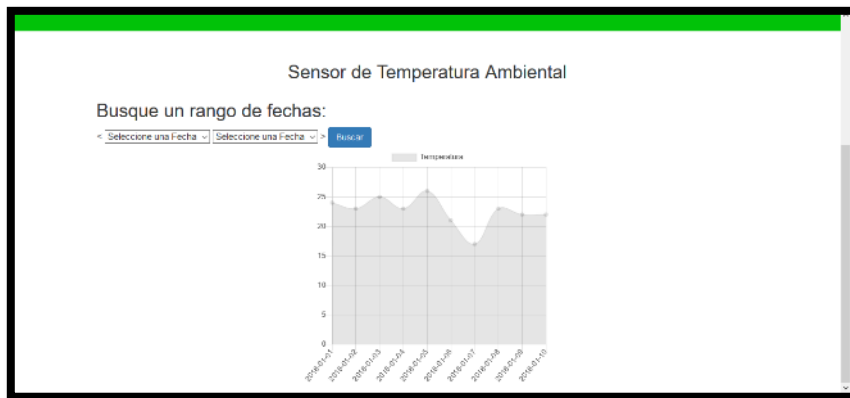


Figura 121. Interfaz web: Gráfico de datos de sensor

Fuente: [El Autor]

### 6.1.2. Usuario administrador

Para acceder a la interfaz de administración se lo realizo a través de un *login*, por lo que al dar clic en el botón “Log in” que se encuentra en la barra de menú de la interfaz web, se desplego un cuadro en el que nos permite ingresar nuestro correo electrónico y contraseña guardadas previamente en nuestra base de datos.

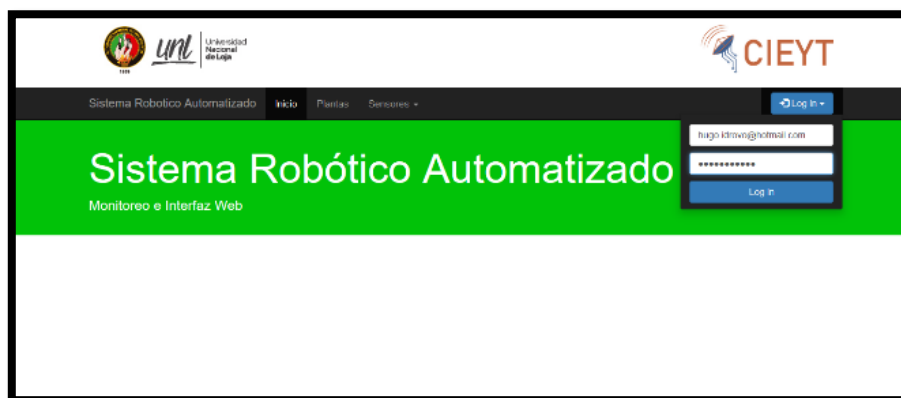


Figura 122. Interfaz web: Cuadro de login

Fuente: [El Autor]

La interfaz web del usuario administrador no difiere mucho de la del usuario invitado por lo que desde aquí también se puede acceder a la información de las plantas y a los datos recogidos de los sensores; a dichos botones se agrega nueva opción de “Control”

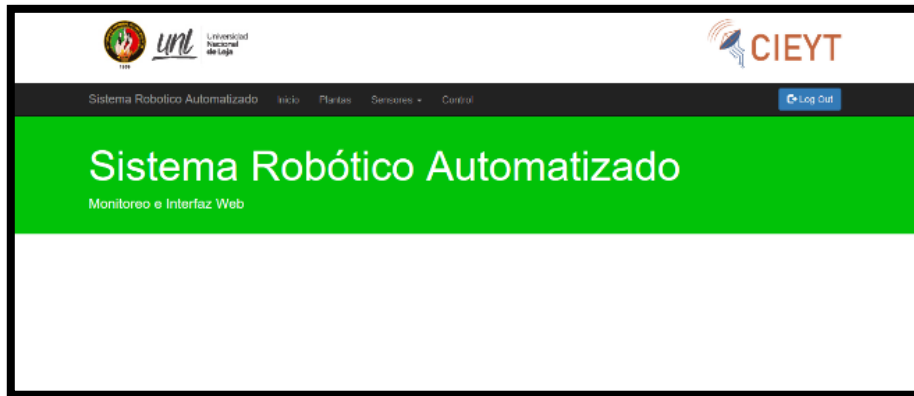


Figura 123. Interfaz web: Página de inicio usuario administrador

Fuente: [El Autor]

Al dar clic en el botón de control se nos abre la interfaz web de control en la que se tiene dos pestañas diferentes, la pestaña “Configuración inicial” cuenta con un selector de plantas, tres cuadros de *check* y un botón de cargar.

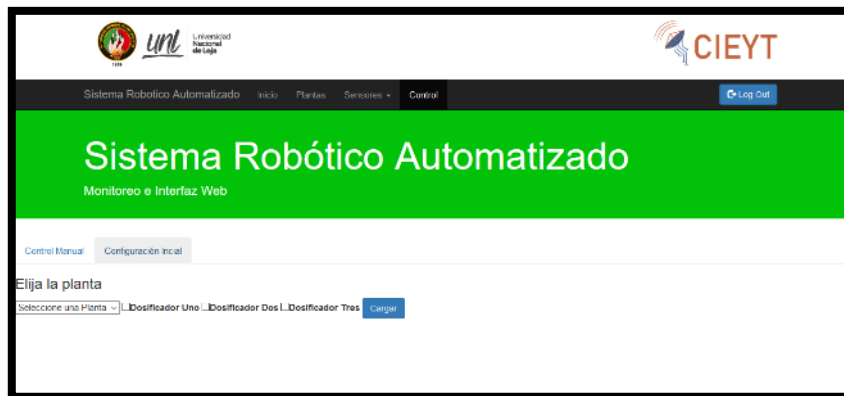


Figura 124. Interfaz web: Configuración inicial

Fuente: [El Autor]

Al darle clic al cuadro de selección se despliega una lista con todas las plantas del sistema, dicha planta seleccionada es la que el sistema registrara en la base de datos, también se debe seleccionar cual o cuales de los tres dosificadores se van a ubicar las semillas a plantar, una vez seleccionadas estas opciones y las semillas ya se encuentren en el/los dosificadores se debe presionar el botón “Cargar” para que empiece el proceso de siembra del sistema.

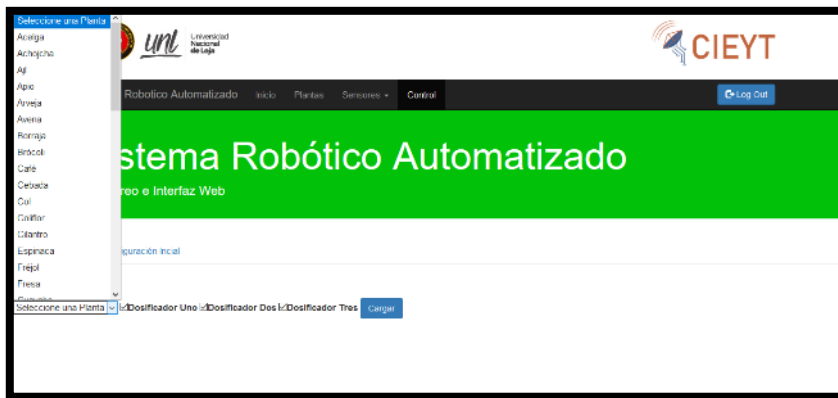


Figura 125. Interfaz web: Configuración inicial selección de planta y dosificador

Fuente: [El Autor]

Una vez cargada la información la interfaz web mostrara en la pestaña “Configuración inicial” la planta que ha sido seleccionada junto con los dosificadores elegidos, adicional a esto se encuentra un botón “Finalizar Sistema” que junto con el uso del *checkbox* nos permite reiniciar el sistema para sembrar otra planta.

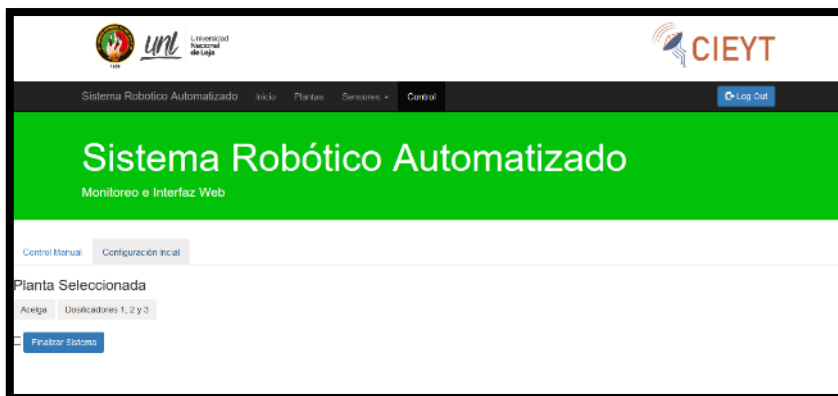


Figura 126. Interfaz web: Configuración inicial completa

Fuente: [El Autor]

En la pestaña “Control Manual” tenemos un conjunto de botones que nos ayudan a controlar el sistema de manera manual, para activar este control manual se debe dar clic en el botón “On”, una vez esto el programa de automatización se encarga de ejecutar todas las acciones que se presionen en la interfaz web, desde sensar y regar hasta mover el brazo robótico según crea conveniente el usuario, cabe resaltar que el sistema detectará cada acción un clic a vez, es decir, si se mantiene presionado un botón este solo ejecutará una operación, mientras el programa no la haya terminado, no se podrá realizar otra.



Figura 127. Interfaz web: Control Manual

Fuente: [El Autor]

En lo que respecta al movimiento de la máquina de manera manual en la interfaz web, se cuenta con cuadros de selección de velocidad y distancia de movimiento, estas se las usa junto con las botones de flechas, el programa detectara y moverá el brazo el trayecto designado y con la rapidez escogida.

## 6.2. Sistema automatizado

El sistema automatizado consiste en un sistema de movimiento en tres ejes, es decir, X, Y y Z, en un sistema para siembra de semillas, a través de un arado especial y unos dosificadores de semillas, un sistema de riego, con la ayuda del llenado de un balde con agua y posterior transporte hasta la ubicación de la semilla/planta, un sistema de almacenamiento, usando para ello bases de datos locales, guardando en ello mediciones de sensores e información especial para funcionamiento del programa de automatización, este último es el que dirige cada acción del sistema en base a la información especial guardada en la base de datos y el archivo de interconexión con la interfaz web.

Al unir todos estos sistemas logramos que nuestro brazo robótico siembre semillas, las riegue y recoja datos de los sensores y los almacene en la base de datos para que el usuario administrador del sistema mediante la interfaz web pueda analizar ciertos factores que afectan el crecimiento de las plantas como son temperatura, humedad ambiental y humedad de tierra, y con ello pueda actuar en consecuencia para que la semilla crezca óptimamente y se obtengan finalmente plántulas de buena calidad.



Figura 128. Sistema automatizado

Fuente: [El Autor]

### 6.2.1. Siembra de semillas

El proceso de siembra de semillas se inicia una vez la planta y el/los dosificadores sean elegidos por el usuario administrador en la configuración inicial de la Interfaz Web, luego de lo cual programa hará uso del sistema de movimiento para ubicarse en la primera cuadrícula, bajar el arador crear con él un agujero soltar semillas con el dosificador subir el arador y moverse a la siguiente cuadrícula y repetir el proceso hasta terminar de sembrar las semillas en todas las cuadrículas.



Figura 129. Sistema de Siembra

Fuente: [El Autor]

### 6.2.2. Riego

El proceso de riego se inicia una vez que las semillas estén sembradas y luego según las mediciones del sensor de humedad de tierra cada dos horas discriminará si es necesario regar, por lo tanto el programa de automatización hará uso del sistema de movimiento para colocarse cerca del centro de la primera cuadrícula, luego bajará el sensor de

humedad de tierra recogerá la medición del sensor y determinará si es una humedad muy baja, en caso de darse esto, riega la semilla/plántula, después se desplaza a las siguiente cuadrícula y la regará, y así hasta completar toda la parcela.



(a)

(b)

Figura 130. a) Comprobación de humedad de tierra; b) Riego

Fuente: [El Autor]

### 6.2.3. Almacenamiento de información (base de datos)

#### 6.2.3.1. Sensor de temperatura/humedad ambiental

El programa de automatización está diseñado para que haga mediciones cada cinco minutos por lo que el programa ayudándose primeramente de la librería MySQL realizará un *query* para actualización de la hora actual y luego otro *query* para consulta a la base de datos solicitando la hora de la última medición, si esta supera el valor establecido de mediciones entonces ayudándose de la librería Serial enviara una solicitud de medición (*string* pre configurado) por USB-Serial hacia el Arduino Uno, encargado de recoger los datos de los sensores, el programa en el Arduino al recibir la solicitud (*string* pre configurado) leerá el dato del sensor y luego le enviara nuevamente por Serial-USB al programa de automatización en la Raspberry, con dicho dato el programa sirviéndose de la librería MySQL procederá a crear el *query* para insertar y lo enviará hacia la base de datos y en la tabla correspondiente del sensor, humedad o temperatura ambiental, y también enviar otro *query* para actualización de la hora de medición, este proceso se repetirá para cada sensor hasta haber hecho la medición de ambos.

idHum_Am	Fecha_Hora	Hum_Amb
1	2018-01-01 00:00:00.0	9
2	2018-01-02 00:00:00.0	12
3	2018-01-03 00:00:00.0	22
4	2018-01-04 00:00:00.0	28

Figura 131. Ejemplo de tabla Humedad Ambiental en base de datos

Fuente: [El Autor]

### 6.2.3.2. Sensor de humedad de tierra

Primero el programa de automatización ayudándose de la librería MySQL realizará un *query* para actualización de la hora actual y luego otro *query* para consulta a la base de datos solicitando la hora de la última medición, si esta supera el valor establecido de mediciones cada dos horas, dado a que está ligado con el sistema de riego, si se supera este tiempo el programa de automatización hace uso del sistema de movimiento y se ubica cerca del centro de la primera cuadrícula, y luego de haber bajado el sensor y enterrarlo, ayudándose de la librería Serial enviara una solicitud de medición (*string* pre configurado) por USB-Serial hacia el Arduino Uno, encargado de recoger los datos de los sensores, el programa en el Arduino al recibir la solicitud (*string* pre configurado) leerá el dato del sensor y luego le enviara nuevamente por Serial-USB al programa de automatización en la Raspberry, con dicho dato el programa sirviéndose de la librería MySQL procederá a crear el *query* para insertar y lo enviará hacia la base de datos y en la tabla correspondiente del sensor, humedad de tierra, y luego hará un *query* de actualización con la actual hora de medición.

idHum_Tie	Fecha_Hora	Hum_Tie
1	2018-01-01 00:00:00.0	9
2	2018-01-02 00:00:00.0	12
3	2018-01-03 00:00:00.0	22
4	2018-01-04 00:00:00.0	28

Figura 132. Ejemplo de tabla Humedad de tierra en base de datos

Fuente: [El Autor]

## 7. DISCUSIÓN

Tras terminar la investigación queda en evidencia que son muchos los parámetros que necesita una planta tanto para germinar y para crecer, esta pueda desarrollarse en un rango de condiciones, pero al salir de estos el vegetal puede directamente no brotar o tener un crecimiento lento o no tenerlo, el rango de condiciones depende de cada tipo de planta, por lo que si se le da esas condiciones una planta que no esté apta para el ambiente en el que se quiere sembrar esta lo hará sin ningún problema.

Tras poner en funcionamiento el sistema automatizado, pudimos en primera instancia delegar a un robot el proceso de siembra que normalmente es tedioso, también el proceso de riego se benefició con esto porque si bien se podría poner un riego usando aspersores estos desperdician mucha agua en el transcurso, por lo que también se brinda la optimización de uso del líquido vital, quedando para el usuario el análisis de datos, abonamiento periódico del suelo y proceso de extracción de las plántulas.

La placa de desarrollo Raspberry Pi es de bajo costo y fácil de encontrar en el país y cuenta con una buenas características (ver Tabla 5) y al ser Open-Hardware no se necesita de licencia alguna para su uso además de poder instalar en ella un sistema operativo Open como es Raspbian basado en Debian Linux, también al ser pequeña (tamaño de una mano) una vez instalado todo lo necesario solo necesita energía eléctrica, una vez se encienda comenzará a realizar las tareas que se le fueron programadas, es decir, ser servidor web, servidor de base datos, y dirigir a los Arduinos para mover, regar, recoger datos, del sistema automatizado creado

La tarjeta electrónica Arduino Mega también es Open-Hardware bajo costo y buenas características (ver 5.2.5.2), además de que en base a él se han desarrollado varias firmware, una de ella para control de motores mediante códigos G, pudiendo con ello crear máquinas CNC de una manera fácil y rápida, con ayuda de programas para CNC o desarrollando uno propio. El uso de los Arduino Uno y Nano (ver 5.2.5.3 y 5.2.5.4) se debe a que tienen características parecidas al Arduino Mega aunque limitados en el número de entradas/salidas posibles a usar, pero no por ello son despreciables, dado que para lo que fueron usadas cumplen correctamente su cometido.

El gestor de base de datos Mariadb, fue usado para configurar el sistema automatizado en lo que respecta al programa de automatización y en varios aspectos de la interfaz web, en



el programa de automatización se necesitaba acceder a la base de datos para consulta, actualización e inserción de datos, mientras que en la página web era necesario la consulta y la actualización. El software PHPMyAdmin nos permitía visualizar la base de datos a través de un GUI, para mayor facilidad de gestión, en el proceso de instalación de las tablas y datos.

Debido al que el número de sensores usados en la base de datos y al intervalo de medición de cada uno se crearon tres tablas diferentes, esto para que la interfaz web consulte de una forma más sencilla la base de datos y pueda luego crear una gráfica de comportamiento con los datos del sensor, facilitando así el seguimiento de parámetros por parte del usuario administrador, y cualquier usuario invitado que tenga acceso a la página web de la interfaz, esto también se aplica a la base de datos de las plantas con su respectiva tabla de visualización.

La instalación de software necesario requiere un nivel medio-bajo de comandos en el terminal de Raspbian (Debian) y manejo de archivos para hacerlo correctamente, mientras que para la configuración se requiere conocimientos en Apache para habilitación de páginas web y módulos adiciones, en Mariadb y PHPMyAdmin para la configuración de tablas, consulta, actualización, en PHP para habilitar módulos dentro de Apache, CodeBlocks para adjuntar las librerías descargadas y no generar errores de compilación, No-IP para puesta en marcha del sistema.

La interfaz web se creó analizando los elementos necesarios con los que debía contar la para hacer que esta sea más fácil de usar debido a su sencillez pero no por ello pobre en diseño, esto le da al usuario una mejor comodidad para usarla, con respecto al contenido de la interfaz web se resolvió el ubicar solamente los datos de sensores y plantas para que el usuario pueda observar de una manera más fácil mediante una gráfica, para el diseño se usó las herramientas de bootstrap que nos permitían estilizar lo con son barras de menús, menús de plegables, cuadros de *login* y botones.

El programa proveído por No-IP es de suma importancia para que el sistema automatizado tenga salida a internet, el programa se encarga de monitorizar la IP pública asignado por nuestro proveedor de servicios de internet, y se encarga de redirigir el nombre de dominio creado en la página web de No-IP hacia la IP publica cuando un usuario invitado o usuario administrado teclea la *url* de la interfaz web en un navegador de internet, para ello

también se configuró el router suministrado por el proveedor de internet para que cuando a este le llegue una solicitud *HTTP* la destine hacia el servidor web ubicado en la Raspberry Pi.

## 8. CONCLUSIONES

- Son muchos los factores que intervienen en el crecimiento óptimo de las plantas por lo que se realizó una exhaustiva revisión literaria para cubrir todos los parámetros, y a partir de ello realizar una correcta elección de equipos de medición.
- Se realizó la exploración extensa de los posibles sistemas, equipos que cumplan en primera instancia el ser de libre difusión (Open Hardware), y también se vio la posibilidad de que existan en el mercado local y/o nacional, con respecto al software también se buscó que sean Open-Software, esto para evitar la compra de licencias, con ello se logró la construcción de toda la parte lógica del sistema automatizado
- Para el diseño se consideró un armazón metálico que abarque un poco más de la parcela en la que se iba a obtener las plántulas, se discurió diferentes métodos de movimiento para sus tres ejes (X, Y, Z) pero se escogió el diseño mediante pernos ya que es la que más estabilidad y precisión nos podía entregar, por lo que con el diseño terminado (ver Figura 19), la construcción se la realizó siguiendo las pautas que nos entregó el diseño y algunas adaptaciones dadas por el técnico del taller mecánico para su correcta elaboración, para adaptar mejor el hardware de movimiento (actuadores).
- Se hizo uso del software CNC en la placa de desarrollo Arduino Mega, para a partir de él y usando códigos G, lenguaje interpretado por el software, avanzar en la elaboración del código fuente del sistema automatizado.
- Con software de desarrollo web se procedió a la creación de la interfaz web que nos permite tanto el control del sistema, es decir, la inicialización y el manejo manual, así como el monitoreo de variables escogidas anteriormente de una forma más sencilla y visual a través de gráficas de comportamiento, mientras que para conectarlo a la red de internet, se valió de herramientas suministradas de manera gratuita por proveedor de servicios DDNS.

## 9. RECOMENDACIONES

En base a las experiencias, análisis y observaciones brindados por el presente proyecto de titulación, se recomienda:

- Efectuar una revisión bibliográfica para tener los conocimientos necesarios para el correcto manejo de todos y cada uno de los elementos usados para el diseño y construcción del sistema automatizado
- Si se hace uso de la placa de desarrollo Raspberry Pi, procurar adquirir una buena fuente de alimentación, así como una tarjeta SD de alta velocidad de escritura y lectura para que ésta funcione de manera más óptima
- El sistema operativo usado en la Raspberry Pi no es la única variante, por lo que si se necesita se puede descargar la distribución con software recomendado, si la Raspberry usada no tiene mucha potencia de procesamiento o el sistema está muy inestable usar la distribución sin escritorio, aunque se requiere nivel avanzado en manejo de terminal de Linux
- Al conseguir los motores Nema 23 tratar de que su voltaje nominal sea de unos 12 voltios o lo más cercana, esto permitirá utilizar la shield Ramps 1.4 que sirve para aplicaciones CNC y se puede conectar a la placa de desarrollo Arduino Mega y utilizarla en lugar de crear una placa de soporte de drivers en PCB para este.
- Si se tiene problemas de alimentación al conectar varios motores Nema 23 en una solo fuente, se debe usar generadores diferentes para cada uno, y tomar en cuenta los voltajes nominales de los motores para obtener fuentes con voltajes iguales o lo más cercano posible ayudándose de reguladores de tensión ya sean fijos o variables según corresponda
- El dosificador de semilla se lo puede desarrollar de manera diferente ya que el creado tiene la desventaja de que se necesita hacer uno para cada tamaño de semilla, por lo que se puede crear uno para que se adapte a cualquier dimensión
- Para el sembrador de semilla se puede tomar en cuenta la creación de un sistema en forma de aguja, de tal forma que no se necesite el arador para hacer el hueco, y solo se deba introducir el sistema de aguja en la tierra y suelte la semilla.

- Se recomienda revisar los anexos 7 y 8 donde se encuentra el manual de usuario y manual de mantenimiento del sistema automatizado.
- Se recomienda revisar los anexos 9 y 10 donde se encuentra los videos que muestran los procesos de siembra y de riego que efectúa el sistema automatizado.

## 10.BIBLIOGRAFÍA

*¿Cómo afecta el viento en las plantas?* (16 de 06 de 2016). Obtenido de Radiacal Nutrients: <https://bit.ly/2IbYtFA>

*¿Qué es el riego?* (s.f.). Obtenido de Hydro Environment: <https://bit.ly/2ULywCK>

*¿Qué es un gestor de datos y para qué sirve?* (26 de 05 de 2016). Obtenido de <https://goo.gl/rfU4Yf>

*¿Qué significa cnc?* (s.f.). Recuperado el 28 de 05 de 2018, de <https://goo.gl/DRkTDz>

*¿Qué son los actuadores electrónicos?* (14 de 10 de 2013). Recuperado el 28 de 05 de 2018, de <https://goo.gl/cRqj9c>

*Acción de la temperatura sobre la vegetación.* (s.f.). Obtenido de Universidad Politécnica de Madrid: <https://bit.ly/2I8TS74>

*Actuador.* (s.f.). Recuperado el 28 de 05 de 2018, de <https://goo.gl/RrVXvB>

*Adafruit\_Sensor.* (s.f.). Obtenido de GitHub: <https://bit.ly/3aAfcxf>

Alonso, R. (08 de 05 de 2018). *¿Qué es un hosting o alojamiento web?* Obtenido de <https://goo.gl/YBQjEi>

Alvarez-Mozos, J., Casalí, J., González-Audicana, M., & López, J. J. (2005). Estimación de la humedad superficial del suelo mediante teledetección radar en presencia de una cubierta de cereal. *Estudios de la Zona No Saturada del Suelo Vol VII*, 313-318.

*Arduino Mega.* (s.f.). Obtenido de Arduino: <https://bit.ly/37bgQU2>

*Arduino Nano.* (s.f.). Obtenido de Arduino: <https://bit.ly/2RjTndU>

*Arduino Uno.* (s.f.). Obtenido de Arduino: <https://bit.ly/36ofofE>

*balenaEtcher.* (s.f.). Obtenido de <https://bit.ly/3anjyHX>

Barberá, J. (s.f.). *Que es una Interfaz web.* Obtenido de <https://goo.gl/vm8QKe>

*Base de Datos (lógica).* (28 de 01 de 2015). Obtenido de <https://goo.gl/aNwY58>

*BootStrap.* (s.f.). Obtenido de BootStrap: <https://bit.ly/36otNIL>

Camps, A., Corbella, I., Bará, J., Torres, F., Duffo, N., & Vall-llossera, M. (2000). La misión de oportunidad SMOS de la serie Earth Explorer. Radiometría por síntesis de apertura para la medida de la humedad del suelo y la salinidad del océano. *BURAN* (págs. 37-43). Barcelona: Escola Tècnica Superior d'Enginyers de Telecomunicació de Barcelona.

*Climatología*. (s.f.). Obtenido de OAS: <https://bit.ly/2Uk2tKH>

*Computadora*. (s.f.). Recuperado el 28 de 05 de 2018, de <https://goo.gl/1SrjY1>

*Crea un host DDNS gratuito con No-IP*. (s.f.). Obtenido de <https://goo.gl/Ku7nuE>

*Definición de Humedad*. (s.f.). Obtenido de ConceptoDefinicion: <https://bit.ly/2KhPGny>

*Definición de pH*. (s.f.). Obtenido de ConceptoDefinicion: <https://bit.ly/2zKLUcJ>

*Definición de Presión atmosférica*. (s.f.). Obtenido de ConceptoDefinicion: <https://bit.ly/2G42rNd>

*Definición de Temperatura*. (s.f.). Obtenido de ConceptoDefinicion: <https://bit.ly/2UttfAv>

*DHT-sensor-library*. (s.f.). Obtenido de GitHub: <https://bit.ly/2RkRM7C>

*DNS dinámicos (DynDNS): accesos remotos sin complicaciones*. (01 de 09 de 2017). Obtenido de <https://goo.gl/A5yYfW>

Doutel, F. (25 de 01 de 2017). *Guía del Arduinomaníaco*. Recuperado el 28 de 05 de 2018, de <https://goo.gl/CKSrs4>

*Driver L298N*. (s.f.). Obtenido de IbeRobotics: <https://bit.ly/2GfLlfG>

*Driver ULN2003*. (s.f.). Obtenido de NaylampMechatronics: <https://bit.ly/36h160s>

*DRV8825*. (s.f.). Obtenido de Pololu Robotics & Electronics: <https://bit.ly/2Rmrrm0>

*Dynamic DNS – The automated rapid record updates*. (12 de 10 de 2016). Obtenido de <https://goo.gl/sXmDdt>

*El medio ambiente*. (s.f.). Obtenido de FAO: <https://bit.ly/2IxPtKa>

*El pH de las plantas*. (s.f.). Obtenido de Agromática: <https://bit.ly/2Gb1jID>

- Garcia Alcacio, J. J. (26 de 10 de 2017). *Findability: La capacidad que posee un PRODUCTO, SERVICIO o INFORMACIÓN de ser encontrado en INTERNET*. Obtenido de Alcacio: <https://goo.gl/rscRU1>
- Geeetech*. (s.f.). Obtenido de <https://bit.ly/2R92OMV>
- González Robles, M. (19 de 09 de 2012). Evaluación del producto "Contenido en humedad del suelo" de nivel 3 de SMOS proporcionado por el centro de procesado CP34 y desde el punto de vista del usuario (Tesis de maestría). Valencia, Valencia, España: Universitat De València.
- grbl-Mega*. (s.f.). Obtenido de GitHub: <https://bit.ly/38vWW6g>
- Griffith, B. (24 de 01 de 2017). *Cómo elegir la computadora de placa única adecuada*. Recuperado el 28 de 05 de 2018, de <https://goo.gl/kyAykE>
- Guimerans, P. (05 de 05 de 2018). *¿Qué es un sensor? Tipos y Diferencias*. Recuperado el 28 de 05 de 2018, de <https://goo.gl/hteSCi>
- Hassan Montero, Y. (2006). Factores del Diseño Web Orientado a la Satisfacción y No-Frustración de Uso. *Revista Española de Documentación Científica*.
- Hassan, Y., & Martín Fernández, F. J. (2004). Propuesta de adaptación de la metodología de Diseño Centrado en el Usuario para el desarrollo de sitios web accesibles. *Revista Española de Documentación Científica*.
- Hermosilla Galeano, M. E. (2015). *abc color*. Obtenido de Nuevas tecnologías aplicadas a la agricultura: <http://www.abc.com.py/edicion-impresa/suplementos/escolar/nuevas-tecnologias-aplicadas-a-la-agricultura-1418784.html>
- Influencia de la lluvia sobre el desarrollo de las plantas*. (17 de 05 de 2005). Obtenido de ABC: <https://bit.ly/2D1OnTA>
- Influencia de la temperatura ambiental en las plantas*. (s.f.). Obtenido de CANNA Research: <https://bit.ly/2DXPXJ1>
- Introducción a la tecnología CNC*. (28 de 12 de 2015). Recuperado el 28 de 05 de 2018, de <https://goo.gl/LGTwb4>
- Joyanes Aguilar, L. (2008). *¿Qué es una computadora?* McGRAW-HILL.



- Joyanes Aguilar, L. (2008). *Clasificación de las computadoras*. McGRAW-HILL.
- Joyanes Aguilar, L. (2008). *El Software (Los Programas)*. McGRAW-HILL.
- Joyanes Aguilar, L. (2008). *Sistema Operativo*. McGRAW-HILL.
- La importancia de la acidez del pH para tus plantas*. (s.f.). Obtenido de Canna:  
<https://bit.ly/1Wqfqvc>
- Laviea, T., & Tractinsky, N. (2004). Assessing dimensions of perceived visual aesthetics of web sites. *International Journal of Human-Computer Studies*.
- López, B. (05 de 06 de 2018). *Qué es un Hosting Web, tipos de Alojamiento y cuál elegir*.  
 Obtenido de <https://goo.gl/FD62cs>
- Los mejores gestores de base de datos del mercado*. (04 de 02 de 2017). Obtenido de  
<https://goo.gl/24WZJv>
- Marín, R. (16 de 04 de 2019). *Los gestores de bases de datos más usados*. Obtenido de  
<https://goo.gl/XfGhwe>
- Milán, V. (24 de 02 de 2018). *El diseño de la interfaz web*. Obtenido de  
<https://goo.gl/8AiVEV>
- Mysql Connector/C++*. (s.f.). Obtenido de Mysql: <https://bit.ly/30WVAPC>
- Nates, J. (2013). *abc*. Obtenido de abc natura-biodiversidad: <http://www.abc.es/natural-biodiversidad/20130412/abci-agricultura-tecnologia-punta-201304121101.html>
- No-Ip*. (s.f.). Obtenido de No-Ip: <https://bit.ly/36FzkuC>
- Pallás Areny, R. (2004). Tipos de Sensores. En *Sensores y Acondicionadores de Señal* (págs. 6-7). Marcombo.
- Pallás Areny, R. (2004). *Transductores, Sensores y accionamientos*. Marcombo.
- Peery, J. (12 de 09 de 2017). *¿Cómo influye la humedad en la calidad de los cultivos?*  
 Obtenido de Pro-Mix: <https://bit.ly/2I93bEJ>
- Pérez Porto, J., & Gardey, A. (2010). *Definición de Sensor*. Recuperado el 28 de 05 de 2018, de <https://goo.gl/kEavKR>
- Pérez Valdés, D. (26 de 10 de 2007). *¿Qué son las bases de datos?* Obtenido de  
<https://goo.gl/1ENXi4>

Perlman, H. (10 de 04 de 2018). *USGS*. Obtenido de La Ciencia del Agua para Escuelas (Water Science for Schools): <https://on.doi.gov/2iP0cFw>

Plasencia Prado, C. E. (s.f.). *¿Qué es y por qué aprender SQL?* Obtenido de <https://goo.gl/rQtA8Z>

*Qué es Arduino*. (s.f.). Recuperado el 28 de 05 de 2018, de <https://goo.gl/3skNjN>

*Qué es el viento, cómo se produce y los diferentes tipos*. (s.f.). Obtenido de Curiosfera: <https://bit.ly/2UkKsfd>

*Raspberry*. (s.f.). Obtenido de <https://bit.ly/375PZbX>

*Raspberry Pi*. (s.f.). Obtenido de Raspbian: <https://bit.ly/2R9q8KJ>

Reyes Mejia, F. (01 de 04 de 2013). *Factores que intervienen en el crecimiento y desarrollo de las plantas*. Obtenido de Fulreyes3: <https://bit.ly/2WYan9o>

Rodríguez Jiménez, R. M., Capa, Á. B., & Portela Lozano, A. (2004). *Meteorología y Climatología*. Fundación Española para la Ciencia y la Tecnología.

Rosenfeld, L., & Morville, P. (2002). *Information Architecture for the World Wide Web*.

*Sensor de humedad del suelo YL38*. (s.f.). Obtenido de Talos Electronics: <https://bit.ly/2NUhqxH>

*Sensor de Temperatura y Humedad DHT22*. (s.f.). Obtenido de AMG Kits: <https://bit.ly/3aIKH8H>

*Sensores para Domótica e Inmótica*. (s.f.). Obtenido de Uniovi: <https://bit.ly/2TF2pjJ>

*SGBD Comerciales*. (s.f.). Obtenido de Basesdedatos's Weblog: <https://bit.ly/2v4KsV6>

*Shields para Arduino*. (s.f.). Recuperado el 04 de 06 de 2018, de <https://goo.gl/HCbCxf>

*Sistema Gestor de Base de Datos*. (s.f.). Obtenido de <https://goo.gl/yBbHwf>

*Sistemas Operativos*. (s.f.). Recuperado el 28 de 05 de 2018, de <https://goo.gl/TRnLyn>

*Stepper Motor*. (s.f.). Obtenido de Pololu Robotics & Electronics: <https://bit.ly/30Ecpyp>

*The water cycle*. (30 de 09 de 2004). Obtenido de ESA: <https://bit.ly/2I2X6sS>

Turtenwald, K. (01 de 02 de 2018). *Los efectos de la presión atmosférica en las plantas*. Obtenido de Geniolandia: <https://bit.ly/2IaDeE7>

*Vibrador mini Motor.* (s.f.). Obtenido de HiFiSac.: <https://bit.ly/2S7t4Im>

*Wagner.* (s.f.). Obtenido de <https://bit.ly/2Rv16Hr>

*Water cycle: land and atmosphere.* (06 de 01 de 2015). Obtenido de ESA:  
<https://bit.ly/2KbLGEQ>

## 11. ANEXOS

### ANEXO 1. Código del programa de automatización

A continuación se ubica parte del código del programa de automatización, para ver el archivo completo revisar en el segundo CD adjunto:

```
while(resp=="Off"){ //Repetir mientras
    ficheroEntrada.open(file); //Apertura de archivo txt
    getline(ficheroEntrada, cadena); //Obtención de informacion almacenada en
archivo txt
    ficheroEntrada.close(); //Cierre de archivo txt
    if(!cadena.empty()){ //Comprobación de valor en variable
        size_t it1 = cadena.find(","); //Busqueda de indicador en variable
        size_t it2 = cadena.find(",it1+1); //Busqueda de siguiente indicador en
variable
        size_t l1 = cadena.copy(comandc,it1,0); //Copia de valores de un char a otro
comandc[l1]='\0'; //Asignación de fin de valor en indicador de char
comand = comandc; //Asignación de valor en string
        if(it2!=std::string::npos){ //Comprobación de variable no vacia
            size_t l2 = cadena.copy(stepc,it2-it1-1,it1+1); //Copia de valores de un char
a otro
            stepc[l2]='\0';
            step = stepc;
            size_t l3 = cadena.copy(velavc,cadena.size()-it2-1,it2+1);
            velavc[l3]='\0';
            velav = velavc;
        }
        else{size_t l2 = cadena.copy(stepc,cadena.size()-it1,it1+1);
            stepc[l2]='\0';
            step = stepc;
        }
        cout<<"Cadena leida: " << cadena <<endl; //Presentación de valor en pantalla
        cout<<"Comando: " <<comand<<endl;
        cout<<"Step: " <<step<<endl;
```

```

    if(!velav.empty()) cout<<"Velav: "<<velav<<endl; //Comprobación de variable
vacía
    if(comand==entrar){ //Comprobacion de igualdad
        stmt->execute("TRUNCATE TABLE Estado"); //Ejecución de comando
Mysql
        stmt->execute("INSERT INTO Estado (Estado) VALUE ('Off');");
    }
    if(comand==salir){
        stmt->execute("TRUNCATE TABLE Estado");
        stmt->execute("INSERT INTO Estado (Estado) VALUE ('On');");
        fo.open(file,ios::trunc); //Apertura de archivo txt y borrado de información
contenida
        fo.close(); //Cierre de archivo txt
        goto aaa; //Ir a indicador (bandera)
    }
    if(comand=="rc"){
        user_input="$H"; //Asignación de valor
        serial_stream_cnc<<user_input<<endl; //Envío de valor por serial a Arduino
Mega (grbl)
        serial_stream_cnc >> read_string; //Recepción de string desde Arduino Mega
(grbl)
        if(read_string=="ok"){
            cout<<"Homing"<<endl;
        }
        else cout<<"Error";
    }
    if(comand=="sent"){
        user_input="T|";
        serial_stream_auno<<user_input; //Envío de valor por serial a Arduino Uno
        serial_stream_auno >> read_string; //Recepción de string desde Arduino
Mega (grbl)

```

```

        stsql="INSERT  Sensor_Temp_Amb  (Fecha_Hora,Temp_Amb)  VALUE
(NOW(),"+read_string+");"; //Preparacion de comando Mysql, guardado de datos en
base de datos

```

```

        stmt->execute(stsql);    //Ejecución de comando Mysql
    }

```

```

    if(comand=="senha"){

```

```

        user_input="H|";

```

```

        serial_stream_auno<<user_input;

```

```

        serial_stream_auno >> read_string;

```

```

        stsql="INSERT  Sensor_Hum_Amb  (Fecha_Hora,Hum_Amb)  VALUE
(NOW(),"+read_string+");";

```

```

        stmt->execute(stsql);

```

```

    }

```

```

    if(comand=="senht"){

```

```

        zresr=-150; //Asignacion de valor real en eje Z con respecto al sensor de
humedad de tierra

```

```

        zt=zresr+zdisr;    //Diferencia entre el valor real y la diferencia de ubicación

```

```

        zts=to_string(zt); //Transformacion de valor int a string

```

```

        user_input="$J=G21G91Z"+zts+"F15";

```

```

        serial_stream_cnc<<user_input<<endl;

```

```

        serial_stream_cnc >> read_string;

```

```

        if(read_string=="ok"){

```

```

            user_input="?";

```

```

            serial_stream_cnc<<user_input<<endl;

```

```

            do{

```

```

                serial_stream_cnc >> read_string;

```

```

            }while(read_string=="ok");

```

```

            statecnc = read_string.substr(0,read_string.find("|")); //Asignación de valor
desde 0 hasta el indicador en string

```

```

            while(statecnc=="<Jog"){

```

```

                user_input="?";

```

```

                serial_stream_cnc<<user_input<<endl;

```

```

do{
    serial_stream_cnc >> read_string;
    }while(read_string=="ok");
statecnc = read_string.substr(0,read_string.find("|"));
if(statecnc!="<Jog"){
    break;    //Salida del iterador while
}
}
user_input="s|";
serial_stream_auno<<user_input;
do{
    serial_stream_auno >> read_string;
}while(read_string!="ok");
if(read_string=="ok"){
    user_input="M|";
    serial_stream_auno<<user_input;
    serial_stream_auno >> read_string;
    stmt="INSERT  Sensor_Hum_Tie  (Fecha_Hora,Hum_Tie)  VALUE
(NOW(),"+read_string+");";
    stmt->execute(stmt);
    read_string="ok";
    if(read_string=="ok"){
        user_input="S|";
        serial_stream_auno<<user_input;
        do{
            serial_stream_auno >> read_string;
            }while(read_string!="ok");
        if(read_string=="ok"){
            zresr=150;    //Asignacion de valor real en eje Z con respecto al
sensor de humedad de tierra
            zt=zresr-zdisr;    //Diferencia entre el valor real
y la diferencia de ubicación

```

```

        zts=to_string(zt); //Transformacion de valor int
a string
        user_input="$J=G21G91Z"+zts+"F15";
serial_stream_cnc<<user_input<<endl;
serial_stream_cnc >> read_string;
if(read_string=="ok"){
    user_input="?";
    serial_stream_cnc<<user_input<<endl;
    do{
        serial_stream_cnc
>> read_string;

    }while(read_string=="ok");
statecnc =
read_string.substr(0,read_string.find("|"));
while(statecnc=="<Jog"){
    user_input="?";
    serial_stream_cnc<<user_input<<endl;
    do{
serial_stream_cnc >> read_string;

    }while(read_string=="ok");
statecnc =
read_string.substr(0,read_string.find("|"));
if(statecnc!="<Jog"){
    break;
    }
}
}
else{cout<<"Error sensor humedad t 1";}

```



```

    }
    else{cout<<"Error sensor humedad t 2";}
    }
    else{cout<<"Error sensor humeda t 3";}
    }
    else{cout<<"Error sensor humedad t 4";}
    }
if(comand=="reg"){
    user_input="a|";
    serial_stream_auno<<user_input;
    do{
        serial_stream_auno >> read_string;
    }while(read_string!="ok");
    if(read_string=="ok"){
        digitalWrite(BA,HIGH);
        delay(8000);
        digitalWrite(BA,LOW);
        read_string="ok";
        if(read_string=="ok"){
            user_input="A|";
            serial_stream_auno<<user_input;
            do{
                serial_stream_auno >> read_string;
            }while(read_string!="ok");
        }
        else{cout<<"Error reg 1";}
    }
    else{cout<<"Error reg 2";}
}

```

## **ANEXO 2. Códigos de la interfaz web**

A continuación, se ubica parte del código de la interfaz web, concretamente la interfaz de información de plantas, para ver el archivo completo revisar en el segundo CD adjunto:

```

<section class="jumbotron">
    <div class="container">
        <h1>Sistema Robótico Automatizado</h1>
        <p>Monitoreo e Interfaz Web</p>
    </div>
</section>
<section class="main_container">
    <div class="container">
        <div class="col-md-12"><h2>Busque una planta: </h2>
        <form action="" method="POST">
            <?php
                include('../consulta.php');
                $consul = new Consulta();
                $list = $consul->List_Pl();
            ?>
            <select class="browser-default custom-select"
                name="plselect" id="plselect">
                <option selected>Seleccione una Planta
                </option>
                <?php
                    while ($listp = $list->fetch_array()){ ?>
                    <option value="<?php echo
                        $listp['idPlantas'];?>">
                    <?php echo $listp['Nombre']; } ?></option>
                </select> >
                <button class="btn btn-primary" action="submit">
                    Buscar</button>
            </form>
        </div>
    </div>
    <?php if(!empty($_POST['plselect'])) {
        //require_once('../consulta.php');
    }

```

```

$plse = new Consulta();
$plansel = $plse->Planta($_POST['plselect']);
$plantse = $plansel->fetch_array();
?>

<div class="container">
<table class="table">
  <thead>
    <tr>
      <th scope="col">Nombre</th>
      <th scope="col">Tipo de siembra</th>
      <th scope="col">Dificultad</th>
      <th scope="col">Riego</th>
      <th scope="col">Temperatura Mnima [C]</th>
      <th scope="col">Temperatura Mxima [C]</th>
      <th scope="col">Temperatura Promedio [C]</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td><?php echo $plantse['Nombre'];?></td>
      <td><?php echo $plantse['Tipo_de_siembra'];?>
      </td>
      <td><?php echo $plantse['Dificultad'];?></td>
      <td><?php echo $plantse['Riego'];?></td>
      <td><?php echo $plantse['Temperatura_Min'];?>
      </td>
      <td><?php echo $plantse['Temperatura_Max'];?>
      </td>
      <td><?php echo $plantse['Temperatura_Prom'];?>
      </td>
    </tr>
  </tbody>

```

```

        </table>
    </div>
    <?php }?>
</section>

```

### ANEXO 3. Archivo de base de datos

Parte del código para importar la base de datos del programa se muestra a continuación, para ver el archivo completo revisar en el segundo CD adjunto:

```

-- Table structure for table `Estado`
DROP TABLE IF EXISTS `Estado`;
/*!40101 SET @saved_cs_client = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `Estado` (
  `idEstado` int(11) NOT NULL AUTO_INCREMENT,
  `Estado` varchar(45) COLLATE utf8_spanish_ci NOT NULL,
  `Sembrado` varchar(45) COLLATE utf8_spanish_ci NOT NULL,
  `Horaact` timestamp(1) NOT NULL DEFAULT CURRENT_TIMESTAMP(1) ON
UPDATE CURRENT_TIMESTAMP(1),
  `Riegoult` timestamp(1) NOT NULL DEFAULT '0000-00-00 00:00:00.0',
  `Thult` timestamp(1) NOT NULL DEFAULT '0000-00-00 00:00:00.0',
  PRIMARY KEY (`idEstado`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8
COLLATE=utf8_spanish_ci;
/*!40101 SET character_set_client = @saved_cs_client */;
-- Dumping data for table `Estado`
LOCK TABLES `Estado` WRITE;
/*!40000 ALTER TABLE `Estado` DISABLE KEYS */;
INSERT INTO `Estado` VALUES (1,'Off','No','2020-01-17 03:18:57.3','2020-01-17
03:16:41.0','2020-01-17 06:58:00.0');
/*!40000 ALTER TABLE `Estado` ENABLE KEYS */;
UNLOCK TABLES;

```

#### ANEXO 4. Programa para arduino uno

A continuación, se ubica parte del código en Arduino IDE para el funcionamiento del Arduino Uno, para ver el archivo completo revisar en el segundo CD adjunto:

```
int paso [4][4] =
{
  {1,1,0,0},{0,1,1,0},{0,0,1,1},{1,0,0,1} //Configuración cuatro pasos para mayor par-
motor
};

int subirS(int dvel){
  for (int i = 0; i < 4; i++)
  {
    digitalWrite(MSIN1, paso[i][0]);
    digitalWrite(MSIN2, paso[i][1]);
    digitalWrite(MSIN3, paso[i][2]);
    digitalWrite(MSIN4, paso[i][3]);
    delay(dvel);
  }
}

int bajarS(int dvel){
  for (int i = 3; i >= 0; i--)
  {
    digitalWrite(MSIN1, paso[i][0]);
    digitalWrite(MSIN2, paso[i][1]);
    digitalWrite(MSIN3, paso[i][2]);
    digitalWrite(MSIN4, paso[i][3]);
    delay(dvel);
  }
}

int pararS(){
  digitalWrite(MSIN1, LOW);
  digitalWrite(MSIN2, LOW);
```

```

    digitalWrite(MSIN3, LOW);
    digitalWrite(MSIN4, LOW);
}
int subirA(int dvel){
    for (int i = 3; i >= 0; i--)
    {
        digitalWrite(MAIN1, paso[i][0]);
        digitalWrite(MAIN2, paso[i][1]);
        digitalWrite(MAIN3, paso[i][2]);
        digitalWrite(MAIN4, paso[i][3]);
        delay(2);
        //58500 revoluciones para subir y/o bajar totalmente
    }
}
int bajarA(int dvel){
    for (int i = 0; i < 4; i++)
    {
        digitalWrite(MAIN1, paso[i][0]);
        digitalWrite(MAIN2, paso[i][1]);
        digitalWrite(MAIN3, paso[i][2]);
        digitalWrite(MAIN4, paso[i][3]);
        delay(2);
        //58500 revoluciones para subir y/o bajar totalmente
    }
}
int pararA(){
    digitalWrite(MAIN1, LOW);
    digitalWrite(MAIN2, LOW);
    digitalWrite(MAIN3, LOW);
    digitalWrite(MAIN4, LOW);
}
int subirP(int dvel){

```

```

for (int i = 0; i < 4; i++)
{
  digitalWrite(MPIN1, paso[i][0]);
  digitalWrite(MPIN2, paso[i][1]);
  digitalWrite(MPIN3, paso[i][2]);
  digitalWrite(MPIN4, paso[i][3]);
  delay(dvel);
}
}

int bajarP(int dvel){
  for (int i = 3; i >= 0; i--)
  {
    digitalWrite(MPIN1, paso[i][0]);
    digitalWrite(MPIN2, paso[i][1]);
    digitalWrite(MPIN3, paso[i][2]);
    digitalWrite(MPIN4, paso[i][3]);
    delay(dvel);
  }
}

int pararP(){
  for (int i = 4; i > 0; i--)
  {
    digitalWrite(MPIN1, LOW);
    digitalWrite(MPIN2, LOW);
    digitalWrite(MPIN3, LOW);
    digitalWrite(MPIN4, LOW);
  }
}

```

## **ANEXO 5. Programa para arduino nano**

A continuación se ubica parte del código en Arduino IDE para el funcionamiento del Arduino Nano, para ver el archivo completo revisar en el segundo CD adjunto:

```
//Secuencia de Motores DC Docificadores
```

```

int velmotor = 10;
int MDUA(int velmot){
  for (int i = 0; i < 4; i++) {
    digitalWrite(M1IN1, HIGH);
    digitalWrite(M1IN2, LOW);
    delay(velmot);
    digitalWrite(M1IN1, LOW);
    digitalWrite(M1IN2, LOW);
  }
}
int MDUC(int velmot){
  for (int i = 0; i < 4; i++) {
    digitalWrite(M1IN1, LOW);
    digitalWrite(M1IN2, HIGH);
    delay(velmot);
    digitalWrite(M1IN1, LOW);
    digitalWrite(M1IN2, LOW);
  }
}
int MDDA(int velmot){
  for (int i = 0; i < 4; i++) {
    digitalWrite(M2IN1, HIGH);
    digitalWrite(M2IN2, LOW);
    delay(velmot);
    digitalWrite(M2IN1, LOW);
    digitalWrite(M2IN2, LOW);
  }
}
int MDDC(int velmot){
  for (int i = 0; i < 4; i++) {
    digitalWrite(M2IN1, LOW);
    digitalWrite(M2IN2, HIGH);

```



```

    delay(velmot);
    digitalWrite(M2IN1, LOW);
    digitalWrite(M2IN2, LOW);
}
}
int MDTA(int velmot){
    for (int i = 0; i < 4; i++) {
        digitalWrite(M3IN1, HIGH);
        digitalWrite(M3IN2, LOW);
        delay(velmot);
        digitalWrite(M3IN1, LOW);
        digitalWrite(M3IN2, LOW);
    }
}
int MDTC(int velmot){
    for (int i = 0; i < 4; i++) {
        digitalWrite(M3IN1, LOW);
        digitalWrite(M3IN2, HIGH);
        delay(velmot);
        digitalWrite(M3IN1, LOW);
        digitalWrite(M3IN2, LOW);
    }
}
//Secuencia de Motores DC Docificadores
int MVUA(){
    digitalWrite(MV1IN1, HIGH);
    digitalWrite(MV1IN2, LOW);
}
int MVUD(){
    digitalWrite(MV1IN1, LOW);
    digitalWrite(MV1IN2, LOW);
}

```

```

int MVDA(){
    digitalWrite(MV2IN1, HIGH);
    digitalWrite(MV2IN2, LOW);
}
int MVDD(){
    digitalWrite(MV2IN1, LOW);
    digitalWrite(MV2IN2, LOW);
}
int MVTA(){
    digitalWrite(MV3IN1, HIGH);
    digitalWrite(MV3IN2, LOW);
}
int MVTD(){
    digitalWrite(MV3IN1, LOW);
    digitalWrite(MV3IN2, LOW);
}

```

## ANEXO 6. Especificaciones técnicas

- **Sensor dht22**

Model	AM2302
Power supply	3.3-5.5V DC
Output signal	digital signal via 1-wire bus
Sensing element	Polymer humidity capacitor
Operating range	humidity 0-100%RH; temperature -40~80Celsius
Accuracy	<b>humidity +-2%RH</b> (Max +-5%RH); temperature +-0.5Celsius
Resolution or sensitivity	humidity 0.1%RH; temperature 0.1Celsius
Repeatability	humidity +-1%RH; temperature +-0.2Celsius
Humidity hysteresis	+0.3%RH
Long-term Stability	+0.5%RH/year
Interchangeability	fully interchangeable

- **Sensor yl-38**

Voltaje de entrada	3.3 – 5 VCD
Voltaje de salida	0 ~ 4.2 V
Corriente	35 mA

Dimensiones	30 x 16 mm
Terminales	
VCC	Tensión de alimentación
GND	Tierra
A0	Salida analógica que entrega una tensión proporcional a la humedad.
D0	Salida digital; este módulo permite ajustar cuándo el nivel lógico en esta salida pasa de bajo a alto mediante el potenciómetro.

- **Sensor de nivel de agua ep2508**

Modelo	IRFP460; TO-247
Wattaje máximo:	10W
Max. Voltaje del interruptor	100V DC
Max. Corriente del interruptor	0.5A
Max. Voltaje de ruptura:	225V DC
Max. Resistencia de contactor	100m Ohm
Rango de temperatura	-10- + 85C
Longitud Total	36mm

#### **ANEXO 7. Manual de usuario**

Para ver el manual de usuario revisar el segundo CD que se encuentra adjunto.

#### **ANEXO 8. Manual de mantenimiento**

Para ver el manual de mantenimiento revisar el segundo CD que se encuentra adjunto.

#### **ANEXO 9. Proceso de siembra**

Para ver el proceso de siembra revisar video respectivo el segundo CD que se encuentra adjunto.

#### **ANEXO 10. Proceso de riego**

Para ver el proceso de riego revisar video respectivo el segundo CD que se encuentra adjunto.