



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL
CARRERA DE INGENIERÍA ELECTRÓNICA**

**“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE ROBOT MÓVIL
DIFERENCIAL ASPERSOR DE ALCOHOL, CON TECNOLOGÍA IOT Y DE
TELEMETRÍA”.**

**TRABAJO DE TITULACIÓN PREVIA A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO ELECTRÓNICO**

AUTOR: KEVIN ANDRÉS FALCONI TAMAYO

ADOLFO DAVID TOMALÁ NAREA

TUTOR: MSC. ING. ORLANDO BARCIA AYALA

GUAYAQUIL-ECUADOR

2022

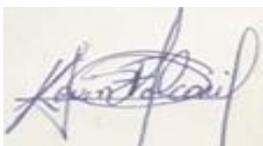
CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Kevin Andrés Falconi Tamayo con documento de identificación N°0930788872 y Adolfo David Tomalá Narea con documento de identificación N°0929359404; manifestamos que:

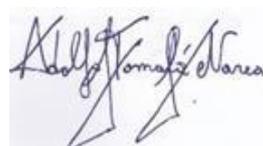
Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Guayaquil, 30 de Septiembre del año 2022

Atentamente,



Kevin Andrés Falconi Tamayo
Ci: 0930788872



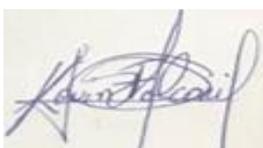
Adolfo David Tomalá Narea
Ci: 0929359404

CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA

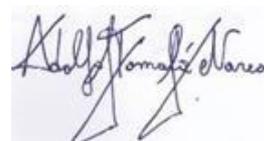
Nosotros, Kevin Andrés Falconi Tamayo con documento de identificación No. 0930788872 y Adolfo David Tomalá Narea con documento de identificación No. 0929359404, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto técnico: “Diseño e implementación de un prototipo de robot móvil diferencial aspensor de alcohol, con tecnología IoT y de telemetría”, el cual ha sido desarrollado para optar por el título de: Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 30 de Septiembre del año 2022

Atentamente,



Kevin Andrés Falconi Tamayo
Ci: 0930788872



Adolfo David Tomalá Narea
Ci: 0929359404

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Orlando Barcia Ayala con documento de identificación N°1309445714, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: “DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE ROBOT MÓVIL DIFERENCIAL ASPERSOR DE ALCOHOL, CON TECNOLOGÍA IOT Y DE TELEMETRÍA”, realizado por Kevin Andrés Falconi Tamayo con documento de identificación N°0930788872 y por Adolfo David Tomalá Narea con documento de identificación N°0929359404, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 30 de Septiembre del año 2022

Atentamente,



ORLANDO
GIOVANNI
BARCIA
AYALA

Ing. Orlando Barcia Ayala, PhD
CI: 1309445714

DEDICATORIA

Dedico este trabajo de titulación a mis conocidos en especial a mi familia que me ha ayudado y han estado conmigo en mis malos y buenos momentos; como a mi padre, madre y hermanos, los cuales me ayudaron en mi proceso de educación hasta la actualidad.

A mis amigos que me ayudaron con sus consejos para que sea una mejor persona en lo laboral como en lo profesional; A mi tía que me ayudó mucho con sus consejos que hay que perseverar y se tiene que luchar y esforzarse por conseguir una meta.

A mis profesores que confiaron en mí y me brindaron su amistad la cual además me enseñaban más de la carrera fuera de sus tiempos laborales.

Y por último agradecer a mi enamorada y su familia por brindarme todo el cariño, amor y confianza de creer en mí y por todos sus consejos de seguir adelante y de hacerme levantar cuando caigo.

Kevin Andrés Falconi Tamayo

DEDICATORIA

Dedico este trabajo de titulación a mis padres por sus grandiosos consejos que me han ayudado a poder terminar este reto, por la paciencia y sabiduría que me han brindado día a día.

A mi mami Rosita y papi Pedro por creer en mi por enseñarme ser perseverante, responsable, orgulloso; la cual me brindo su cariño y amor. Su sueño más grande fue verme titulado que sé que en el cielo lo estará celebrando con todo fulgor.

A mis compañeros y amigos por brindarme su amistad y de ayudarme en los momentos de flaqueza.

A los docentes que supieron guiarme y enseñar nuevas temáticas de estudio para mi futuro profesional en la vida cotidiana.

Adolfo David Tomalá Narea

AGRADECIMIENTO

Agradezco a Dios, a mis padres
A nuestras familias y a los señores
Profesores por el apoyo brindado
Para culminar nuestro sueño anhelado

Kevin Andrés Falconi Tamayo
Adolfo David Tomalá Narea

RESUMEN

AÑO	ALUMNOS	DIRECTOR DE PROYECTO TÉCNICO	TEMA DE PROYECTO TÉCNICO
2022	Kevin Andrés Falconi Tamayo Adolfo David Tomalá Narea	ING: Orlando Giovanni Barcia Ayala, Msc.	Diseño e implementación de un prototipo de robot móvil diferencial aspersor de alcohol, con tecnología IOT y de Telemetría

El robot móvil diferencial aspersor de alcohol con tecnología IOT y de Telemetría, combina soluciones de para tratar de disminuir la carga viral y las bacterias en lugares cerrados como son: las aulas y laboratorios de la institución, sin la necesidad de manipulación humana en la zona de posible alta carga viral.

Cuenta con sensores para medir diversos parámetros para su mejor uso, los cuales son: sensores ultrasónicos, sensores de nivel tipo bolla, además de luces guías las cuales indican procesos como inicio de desinfección (fumigación) y fin de la desinfección; cada uno de los sensores y actuadores son compatibles y adaptables a un microcontrolador que es el Arduino Mega 2560.

El sistema trabaja con internet de las cosas (IoT), el cual nos permite observar y crear un registro de los datos en un servidor gratuito como es el caso de IBM Cloud y Microsoft Azure, el cual nos da un servicio de Node red que es una API para poder comunicar nuestro microprocesador y una cámara IP que estará adaptado, este nos permitirá observar en tiempo real lo que sucede alrededor y a su vez este servicio nos permite crear un dashboard (tablero), la cual nos servirá para control del movimiento del robot por medio de telemetría.

El servicio de Node red estará programado tanto como en la nube del servicio de IBM y Microsoft Azure, como de manera local a través de un miniordenador como es el Raspberry Pi4 para que exista una comunicación bidireccional; La comunicación de nuestro miniordenador (Raspberry Pi4) a nuestro microprocesador (Arduino) es por

medio de puerto serial que facilita el transporte bidireccional de datos.

Este puede ser utilizado como modelo para futuros proyectos de robótica aplicada con tecnología IOT y telemetría en la Universidad Politécnica Salesiana, además se podría ingresar en el campo de la robótica hospitalaria ya que con ello se ve un antes y después en el desarrollo tecnológico del país.

ABSTRACT

DATE	STUDENTS	TECHMICAL PROJECT MANAGER	TECHNICAL PROJECT SUBJECT
2022	Kevin Andrés Falconi Tamayo Adolfo David Tomalá Narea	ING: Orlando Giovanni Barcia Ayala, Msc.	Design and implementation of a prototype mobile alcohol sprinkler differential robot with IOT and Telemetry technology

The differential mobile alcohol sprinkler robot, with IoT technology and telemetry, combines solutions to try to decrease the viral and bacterial load in closed places such as: the classrooms and laboratories of the institution without the need for human manipulation in possible high viral load.

It has to sensors to measure various parameters for better use, which are: ultrasonic sensors, buoy-type level sensors, in addition to guide lights which indicate processes such as the beginning of disinfection (fumigation) and the end of disinfection; each of the sensors and actuators are compatible and adaptable to a microcontroller that is the Arduino Mega 2560.

The system works with the internet of things (IoT), which allows us to observe and create a record of the data on a free server such as IBM Cloud and Microsoft Azure, which gives us a network node service that is an API to communicate our microprocessor and an IP camera that will be adapted, this will allow us to observe in real time what is happening around us and in turn this service allows us to create a dashboard, which will serve us to control the movement of the robot through telemetry.

The red node service will be programmed as well as in the cloud service of IBM and Microsoft azure, as well as locally through a minicomputer such as the Raspberri Pi4 so that there is bidirectional communication. The communication from our minicomputer (Raspberry Pi 4) to our microprocessor (Arduino) is through a serial port that facilitates the bidirectional transport of data.

This can be used as a model for future applied robotics projects with IOT technology and telemetry at the Salesian Polytechnic University. In addition, it would be possible to enter the field of hospital robotics, since with this a before and after can be seen in the technological development of the country.

ÍNDICE GENERAL

TABLA DE CONTENIDO

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN	ii
CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA.....	iii
CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN.....	iv
DEDICATORIA	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
RESUMEN.....	viii
ABSTRACT	x
ÍNDICE GENERAL	xii
INTRODUCCIÓN.....	1
1. PROBLEMA.....	2
1.1. PLANTEAMIENTO DEL PROBLEMA.....	2
1.2. ANTECEDENTES.....	3
1.3. IMPORTANCIA Y ALCANCE.....	3
1.4. DELIMITACIÓN DEL PROBLEMA.....	4
1.4.1. DELIMITACIÓN TEMPORAL.....	5
1.4.2. DELIMITACIÓN ESPACIAL.....	5
1.4.3. DELIMITACIÓN ACADÉMICA	5
1.5. EXPLICACIÓN DEL PROBLEMA	5
1.6. OBJETIVOS	6
1.6.1. OBJETIVOS GENERAL	6
1.6.2. OBJETIVO ESPECIFICO	6
1.7. JUSTIFICACIÓN	7
2. FUNDAMENTACIÓN TEÓRICO.....	8
2.1. ANTECEDENTES.....	11
2.2. ROBOT MÓVIL.....	11
2.2.1. ROBOT MÓVIL CON PLATAFORMA DE RUEDAS DIFERENCIAL	11
2.3. RASPBERRI PI	12
2.4. ARDUINO.....	12

2.5.	NODE RED.....	2
2.6.	IBM CLOUD.....	3
2.7.	IoT.....	3
2.8.	CLIENTE/SERVIDOR.....	4
2.9.	DASHBOARD.....	4
2.10.	DESINFECCIÓN.....	4
2.11.	TELEMETRÍA.....	5
2.12.	SENSOR ULTRASÓNICO.....	5
2.13.	DRIVER.....	6
2.14.	CÁMARA IP.....	7
2.15.	SENSOR DE NIVEL TIPO BOYA.....	7
2.16.	CARGA BACTERIANA.....	8
2.17.	NODE.JS (JASON).....	8
2.18.	ARCHIVO.JS.....	9
2.19.	INDUSTRIA 4.0.....	9
2.20.	CLOUD IBM VS OTROS CLOUD.....	9
2.21.	BARE METAL.....	10
2.22.	MICROSOFT AZURE.....	10
2.23.	OBS STUDIO.....	10
2.24.	ROBOTS FUMIGADORES EN OTROS PAÍSES.....	11
3.	MARCO METODOLÓGICO.....	13
3.1.	DESCRIPCIÓN DEL PROTOTIPO.....	13
3.2.	ETAPAS DEL PROTOTIPO.....	14
3.3.	ETAPA DE PROCESAMIENTO.....	14
3.4.	ETAPA DE ADQUISICIÓN.....	15
3.5.	ETAPA DE APLICACIÓN.....	16
3.6.	DIAGRAMA DE CONEXIONES.....	16
3.7.	CONEXIÓN TARJETA DE POTENCIA.....	16
3.8.	CREACIÓN DE CUENTA IBM.....	17
3.9.	IBM WATSON IOT.....	19
3.10.	MICROSOFT AZURE.....	23
3.11.	NODE RED.....	33
3.12.	ARDUINO ID.....	61
3.13.	PROTEUS.....	64
3.14.	AUTOCAD.....	66
4.	RESULTADOS.....	67

4.1. ANÁLISIS DE RESULTADOS	69
5. CONCLUSIONES	70
6. RECOMENDACIONES.....	71
Anexos	77
Proceso de montaje del robot.....	0
MANUAL DE USUARIO	4
DASHBOARD:	5
MANUAL TÉCNICO	7
PLANOS:	7
Estructura del robot	7
Diseño placa de distribución para Arduino Mega	10
Diseño de placa de distribución de indicadores leds.....	12
DISEÑO DE PLACAS DE DISTRIBUCIÓN DE ACTUADORES Y SENSORES...	13
COMPONENTES ELECTRÓNICOS:	15
Motor reductor de metal 19: 1 37Dx52L mm 12V (piñón helicoidal)	15
8GB E14 Version Raspberry Pi 4 Model 4B BCM2711 Quad-core Cortex-A72 1.5GHz 8GB RAM with Dual Band WIFI Support POE.....	15
Pololu Dual VNH5019 Escudo del controlador del motor para Arduino	16
Bomba de agua dulce de 12 V CC con 2 abrazaderas de manguera Bomba de diafragma de 12 voltios Bomba de rociador autocebante con interruptor de presión 4.5 L/Min 1.2 GPM 80 PSI ajustable para RV Camper Marine Boat Lawn	19
Turnigy Graphene Panther 6000mAh 3S 75C Batería w / XT90	19
Regulador de voltaje reducido del convertidor de CC LM2596S ajustable módulo de fuente de alimentación 36V 24V 12V a 5V 2ª Estabilizador de voltaje con pantalla digital voltímetro (6)	20
Sensor Horizontal de Nivel de Agua – Flotador Interruptor	21
Arduino Mega 2560	22
Sensor ultrasónico HC-SR04.....	22

ÍNDICE DE IMÁGENES

Figura 1. Designación de espacio del proyecto técnico (Google maps).....	5
Figura 2. Estructura de un robot diferencial	12
Figura 3. Ordenador Raspberry.	12
Figura 4. Microprocesador Arduino Mega.	2
Figura 5. Esquema de programación Node red.	2
Figura 6. Logotipo IBM Cloud	3
Figura 7. Conectividad por IoT	4
Figura 8. Sensor ultrasónico de Arduino	6
Figura 9. Tarjeta driver para led de tensión constante o corriente constante.	6
Figura 10. Cámara IP	7
Figura 11 Sensor de nivel tipo boya	8
Figura 12 Diagrama de bloques del robot aspersor.....	13
Figura 13. Etapas del prototipo	14
Figura 14. Etapa de procesamiento	15
Figura 15. Etapa de adquisición.....	15
Figura 16. Etapa de aplicación.....	16
Figura 17 Ingreso a IBM Cloud para creación de cuenta.....	18
Figura 18. Creando servicio IoT.....	18
Figura 19. Creando el servicio Watson IoT.....	19
Figura 20. Agregando un dispositivo a IBM Watson.....	20
Figura 21. Identificando al nuevo dispositivo IoT	20
Figura 22. Dando seguridad a la señal IoT	21
Figura 23. Generando comunicación bidireccional de IoT a node red	21
Figura 24. Generando información a la clave API.....	22
Figura 25. Generando un rol de acceso a la API	22
Figura 26. Lista de recursos de IBM Cloud.....	23
Figura 27. Ingresando a Microsoft Azure.....	24
Figura 28. Seleccionando método de pago para Azure	24
Figura 29. Creando máquina virtual	25
Figura 30. Ingreso a los datos básicos de la máquina virtual	25
Figura 31. Detallando la instancia.....	26
Figura 32. Creando usuario y contraseña a la máquina virtual.....	26
Figura 33. Selección de recursos para configuración de la MV.....	27
Figura 34. Obteniendo la IP pública de la MV	27
Figura 35. Configurando la IP publica en el administrador de sistema operativo	28
Figura 36. Seleccionando el host remoto del SSH.	28
Figura 37. Ingresando a la máquina virtual.....	29
Figura 38. Rutinando el servidor de la MV	29
Figura 39. Instalando comandos para activación de la MV	30
Figura 40. Rebotando la MV y finalizando el proceso activación	31
Figura 41. Comprobando el proceso de activación.....	31

Figura 42. Configurando la regla de seguridad de entrada	32
Figura 43. Ingresando y visualizar la MV.....	32
Figura 44. Máquina virtual en Azure.....	33
Figura 45. Generando usuario y contraseña del Node red.....	33
Figura 46. Ingresando a node red.....	34
Figura 47. Ingresando al editor de Node red	34
Figura 48. Ingresando al administrador de paleta de Node red.....	35
Figura 49. Instalando IBM Watson al Node red.....	36
Figura 50. Características del nodo instalado.....	36
Figura 51. Instalando Node IBM IOT a nuestro Node red	37
Figura 52. Verificación de instalación de nodos	38
Figura 53. Entrando a la lista de recursos de IBM Cloud.	38
Figura 54. Ingreso al servicio IoT IBM.	39
Figura 55. Ejecutando IBM Watson IOT platform.....	40
Figura 56. Plataforma IBM Watson y selección de la organización.	40
Figura 57. Seleccionando Node Watson IOT de salida al scrip de Node red.....	41
Figura 58. Editando Node Watson IoT.....	42
Figura 59. Editando credenciales del nodo Watson IoT.	43
Figura 60. Selección de nodo button para el dashboard.	43
Figura 61. Editando nodo button la propiedad del grupo.....	44
Figura 62. Descargando librería de iconos para los nodos button del dashboard.....	45
Figura 63. Editando las demás opciones del node button del dashboard.....	45
Figura 64. Editando nodo button el tipo de variable.	46
Figura 65. Editando el nodo button (dando valor alfabético y nombre a la variable). ...	47
Figura 66. Seleccionando el nodo msg.payload.	48
Figura 67. Conexión de los nodos buttons al nodo Watson IoT	49
Figura 68. Selección de nodo template para comunicación de cámara IP.	49
Figura 69. Configuración del nodo template	50
Figura 70. Ingresando al dashboard.	51
Figura 71. Editando el dashboard.....	51
Figura 72. Configurando el tab del dashboard	52
Figura 73. Creación de grupos en el tab del dashboard.....	52
Figura 74. Ingreso de buttons en grupos e ingreso de diseño al dashboard.....	53
Figura 75. Diseño de la estructura del dashboard.	54
Figura 76. Verificación de caracteres enviados de node red a IBM Watson IoT.	54
Figura 77. Selección de nodo IBM IoT.....	55
Figura 78. Configuración del nodo IBM IoT seleccionando el tipo de autenticación.	56
Figura 79. Configuración de nodo IBM IoT con la API Key.	56
Figura 80. Configuración del nodo IBM IoT (selección del token y clave).	57
Figura 81. Configuración del nodo IBM IoT (selección del tipo de dispositivo y formato)	57
Figura 82. Configuración del nodo switch	58
Figura 83. Configuración del nodo de funciones.....	58
Figura 84. Escogiendo nodo de comunicación serial.	59
Figura 85. Editor del nodo de salida de serial.	59

Figura 86. Configurando el número de bit y número de puerto en el nodo serial	60
Figura 87. Esquema del nodo red de la configuración del nodo IBM IoT	60
Figura 88. Declaración de las variables e arduino ID	62
Figura 89. Asignación de letras a nuestros movimientos y velocidades	63
Figura 90. Asignación de métodos para el giro de los motores	64
Figura 91. Esquema electrónico en Proteus	65
Figura 92. Diseño de PCB en Proteus	65
Figura 93. Diseño de planos en Autocad.....	66
Figura 94. Placa de distribución de sensores y baliza	67
Figura 95. Tablero o dashboard.....	68
Figura 96. Transmisión de la cámara IP a YouTube	68
Figura 97. Proceso de dobles y cortada de la estructura.....	0
Figura 98. Proceso de limar las imperfecciones de la estructura del robot	0
Figura 99. Proceso de empaste de la estructura del robot móvil.....	1
Figura 100. Ensamblaje de las placas del robot fumigador.....	1
Figura 101. Montaje de actuadores y tarjetas electrónicas a la estructura del robot	2
Figura 102. Diseño final del robot fumigador	4
Figura 103. Controles del dashboard.....	5
Figura 104. Vista superior de la estructura del robot fumigador	7
Figura 105. Vista inferior de la estructura del robot fumigador	8
Figura 106. Vista del contorno del robot fumigador	8
Figura 107. Vista Este y Oeste del robot fumigador	9
Figura 108. Vista Norte y Sur del robot fumigador.....	9
Figura 109. Vista en 3D superior e inferior del robot fumigador	10
Figura 110. Vista inferior de placa de distribución del Arduino Mega.....	10
Figura 111. Vista superior de placa de distribución del Arduino Mega.....	11
Figura 112. PCB de la placa de distribución del Arduino Mega.....	11
Figura 113. Vista superior de la placa de distribución de indicadores leds.	12
Figura 114. Vista inferior de la placa de distribución de indicadores leds.	12
Figura 115. PCB de la placa de distribución de indicadores leds.....	13
Figura 116. Vista superior de la placa de actuadores y sensores.....	13
Figura 117. Vista inferior de la placa de actuadores y sensores	14
Figura 118. PCB de la placa de actuadores y sensores	14
Figura 119. Dimensiones del motor reductor.....	15
Figura 120. Conexión entre el arduino y driver de motor.....	17
Figura 121. Diagrama esquemático del driver del motor	18
Figura 122. Bomba de fumigación.....	19
Figura 123. Fuente de alimentación (batería tipo grafeno).....	19
Figura 124. Regulador de voltaje.....	20

ÍNDICE DE TABLAS

Tabla 1. Variación en casos por reclasificación Ecuador- Covid 1- 2020 [25].....	10
Tabla 2. Entradas de la tarjeta de potencia.....	17
Tabla 3. Salidas de la tarjeta de potencia	17
Tabla 4. PRESUPUESTO PARA LA CONSTRUCCIÓN DE UN ROBOT MÓVIL DIFERENCIAL ASPERSOR DE ALCOHOL, CON TECNOLOGÍA IOT Y TELEMETRÍA	2

INTRODUCCIÓN

Las enfermedades causadas por agentes patógenos transmitidos por microorganismos causado por transmisión de un receptor a otro, por falta de higiene o por mala desinfección de área altamente contagiosas, es un mal que hasta la actualidad existe. Por lo general la mayor parte de la población se contagia por el mal proceso de higiene, que se debe al mal uso de equipos de desinfección e incluso la mayoría siguiendo el debido protocolo no posee una protección total del 100% de contraer un agente patógeno. Esto quiere decir que una gran parte de contagios se debe a la presencia del ser humano en zonas con alta carga bacteriana.

En 2021, el país atravesó un proceso de pandemia que fue a nivel mundial, provocando muchas pérdidas a nivel económico como nivel familiar, la cual llevo a la población al confinamiento por casi 6 meses y trabajos semipresenciales.

Para ello el presente proyecto de titulación expone el diseño de un prototipo de robot móvil diferencial que posee un aspersor de alcohol, que está enfocado en desinfectar (fumigando) espacios cerrados como laboratorios, edificios con departamentos, hospitales, oficinas, centros comerciales; la cual será guiado por medio de cualquier dispositivo que posea internet por medio de una API. Este prototipo surge de la necesidad de realizar el proceso de desinfección de manera segura para el personal de limpieza.

El proyecto está dirigido a estudiantes de la carrera electrónica de la Universidad Salesiana de Guayaquil, la cual ha sido realizado bajo la supervisión del Ing. Orlando Barcia docente que está a cargo de la revisión de este trabajo de titulación, que servirá para implementar los laboratorios de practica y fortalecer la didáctica y pedagogía para los conocimientos de los alumnos que sigan esta especialidad (robótica).

1. PROBLEMA

1.1. PLANTEAMIENTO DEL PROBLEMA

Actualmente a nivel mundial las enfermedades representan una gran pérdida a nivel económico como personal, la cual por enfermedades ET (enfermedades transmisibles) aproximadamente el 49.4% de la población total se infectó por el brote de la pandemia actual conocida como Covid-19. La presente situación provocó una crisis económica del mercado y amenazas para las pequeñas y medianas empresas (PIMES) debido que manejan baja producción y rendimiento. [21]

Su forma de propagación según la OMS y el Ministerio de Salud Pública, es conocida que es por vía aérea transmitiéndose por una partícula de saliva de un emisor a un receptor, se conoce que las pandemias de acuerdo a su naturaleza han causado varios problemas de diferentes índoles en los seres humanos, por lo cual ha sido necesario la desinfección de todos los lugares cerrados contaminados y no contaminados para evitar la propagación de las mismas.

Para lograr mejorar esta situación, se plantea la posibilidad de que el labor de desinfección se encargue de realizarlo un agente externo en lugar de una persona, la cual habrá una disminución de riesgos de salud que representa la exposición de una carga bacteriana alta, por medio de la introducción de nuevas tecnologías, se plantea el desarrollo de un robot móvil capaz de llevar a cabo el proceso de desinfección de forma teledirigida, diseñado especialmente para las áreas de mayor carga viral. Necesariamente el personal deberá cumplir las normas de bioseguridad para evitar la propagación, el contagio siempre existirá y será más fácil entre los seres humanos por eso se vio la necesidad de la creación de una maquina razón suficiente para la aplicación en la biomedicina.

1.2. ANTECEDENTES

Existen muchas zonas infectadas por varios agentes patógenos, en muchas áreas sea uso público o privado. En la desinfección las personas que realizan la limpieza, para poder habilitar los espacios que han sido contagiados con carga viral o bacteriana, y desean obtener un porcentaje casi 0 de desinfección son entes con mayor probabilidad de contagiarse, y pueden sufrir resquebrajamiento en su salud por inhalar de forma involuntaria los distintos agentes químicos, a causa de los ciclos repetitivos y de llevar el virus o bacterias a sus familiares.

Para que no exista un porcentaje alto de un posible riesgo de contagio en las personas que efectúan la limpieza o desinfección de las zonas, se crea este prototipo de robot móvil, aspersor de alcohol que se opera vía telemática, para evitar que dicho personal no tenga contacto con las diferentes zonas de riesgo. Con la ayuda de una cámara integrada, el operador observa en vivo, el camino que el robot móvil vaya a efectuar para la desinfección.

De igual manera se reduce los desechos de equipos de protección personal de Bioseguridad como son las mascarillas, guantes y visores que comúnmente son para un solo uso. Porque el operador estaría realizando su trabajo desde un lugar seguro, mientras que el robot diferencial, aspersor de alcohol, estaría combatiendo en primera línea las zonas de menor a mayor carga viral o bacteriana.

1.3. IMPORTANCIA Y ALCANCE

Este proyecto impulsa a la robótica en el Politécnico Salesiano de Guayaquil, teniendo en cuenta los problemas actuales de limpieza de salones, pasillos, laboratorios, oficinas, etc. EL autor observo que en la sede en trámites de pago o solicitudes donde se aglomeran un gran grupo de personas hay una mayor carga viral.

Los protocolos de seguridad para una carga viral alta son: Uso de mascarilla, alcohol antiséptico, carnet de vacunación, distancias de 6 pies de los demás, espacios con

multitudes y poca ventilación, limpieza y desinfección. El procedimiento de estos protocolos es responsabilidad de cada uno del personal externo e interno del establecimiento, según lo establece la ONU o el Ministerio de Salud Pública.

Los principales beneficiarios son: los estudiantes, docentes, personales administrativos, personal de limpieza y los establecimientos donde se realizan los diferentes movimientos diarios para adquirir lo necesario en vestimenta, alimentación, medicamentos, etc.

El alcance que tiene involucra a todo el establecimiento de la Universidad, para la desinfección de áreas que haya una alta carga bacteriana que puede producir enfermedades. Se debe considerar que este prototipo es de un uso para áreas cerradas a $45 m^2$ aproximadamente.

El impacto que se debe obtener con la implementación del robot móvil diferencial, aspersor de alcohol, es el dar a conocer a las personas que se puede combatir una pandemia el día de hoy, con la tecnología actual utilizando robots teledirigidos, la cual no se hace participe la presencia de un ser humano en el área de contaminación y evitar más contagios. Así mismo, deseamos tener un impacto positivo para las entidades públicas, como los hospitales, centros comerciales y sitios de uso masivo para combatir esta epidemia.

1.4. DELIMITACIÓN DEL PROBLEMA

El robot móvil diferencial que se está implementando con su diseño es un prototipo que contiene un aspersor de alcohol, con tecnología IOT y de Telemetría”, se delimitará para uso de dicha institución, así como también para el manejo de espacios cerrados como aulas, oficinas, centros comerciales, edificios; esto se dará con la ayuda y operado por internet a través de una red la cual por ese medio va a ser tele operado para su movimiento y acciones de fumigación.

1.4.1. DELIMITACIÓN TEMPORAL

Este proyecto de título se completó de 5 meses y se finalizó en el actual año 2022

1.4.2. DELIMITACIÓN ESPACIAL

Este proyecto estará ubicado en el Laboratorio flexible del edificio E de la Universidad Salesiana de Guayaquil, como se muestra en la Figura 1. referenciada en un mapa geográfico de ubicación de la universidad



Figura 1. Designación de espacio del proyecto técnico (Google maps)

1.4.3. DELIMITACIÓN ACADÉMICA

Se efectúa demostraciones para la aplicación de la robótica en diferentes áreas, con el fin de que el alumnado aprenda los diferentes funcionamientos y usos que puede tener este prototipo.

1.5. EXPLICACIÓN DEL PROBLEMA

Muchas personas que hacen el trabajo de limpieza, sufren altos riesgos de contraer un agente patógeno debido a su trabajo, algunas veces por mucha carga bacteriana en ese entorno donde se encuentra o por el uso indebido de materiales de seguridad para estos trabajos.

El problema radica en: ¿cómo se puede hacer este proceso de desinfección de zonas de alto riesgos sin manipulación humana?, para ello, se tiene que usar un dispositivo u objeto, que permita desinfectar los espacios usando el método de fumigación y que un operador lo está manipulando desde una distancia muy razonable, para que pueda visualizar en tiempo real donde fumigar y pueda observar el entorno para evitar colisiones.

En los estudios brindados por la Universidad Politécnica Salesiana en el área de robótica se ha inculcado que existen muchos tipos de robots para diferentes tipos procesos como esta actividad ya mencionada además que esta tecnología se está implementando en otros países y son muy fáciles de armar utilizando microprocesadores para su comunicación con sus actuadores, drivers, sensores. Además, para la obtención de una comunicación más segura y estable se utiliza comunicación IoT (internet de las cosas) para que cualquier dispositivo que tenga acceso a internet pueda controlar el robot desde cualquier lugar por medio de una dashboard (control de mando) amigable al usuario

1.6. OBJETIVOS

1.6.1. OBJETIVOS GENERAL

Diseñar e implementar un prototipo de robot móvil diferencial aspersor de alcohol, con tecnología IoT y Telemetría para la Universidad Politécnica Salesiana.

1.6.2. OBJETIVO ESPECIFICO

- Integrar el microcontrolador, ordenador de placa reducida (Raspberry Pi), sensores, actuadores y cámara para el control del robot diferencial aspersor de alcohol por medio de IoT.
- Elaborar el Dashboard para la telemetría con el operador.
- Integrar una cámara, para que el operador pueda observar en tiempo real los objetos alrededor del robot móvil diferencial.

1.7. JUSTIFICACIÓN

En diferentes zonas que poseen mayor carga viral o bacteriana son: las unidades de cuidados intensivos, los pasillos, salas de especialidades, aulas, edificios y en un proyecto futuro centros comerciales, patios de comida, etc.

En cualquiera de las zonas mencionadas, las personas que realizan la limpieza, para poder habilitar los espacios que han sido contagiados con carga viral o bacteriana, y desean obtener un porcentaje casi 0 de desinfección son entes con mayor probabilidad de contagiarse, y pueden sufrir resquebrajamiento en su salud por inhalar de forma involuntaria los distintos agentes químicos, a causa de los ciclos repetitivos y de llevar el virus o bacterias a sus familiares.

Para que no exista un porcentaje alto de un posible riesgo de contagio en las personas que efectúan la limpieza o desinfección de las zonas, se crea este prototipo de robot móvil, aspersor de alcohol que se opera vía telemática, para evitar que dicho personal no tenga contacto con las diferentes zonas de riesgo. Con la ayuda de una cámara integrada, el operador observa en vivo, el camino que el robot móvil vaya a efectuar para la desinfección.

De igual manera se reduciría los desechos de equipos de protección personal de Bioseguridad como son las mascarillas, guantes y visores que comúnmente son para un solo uso. Porque el operador estaría realizando su trabajo desde un lugar seguro, mientras que el robot diferencial, aspersor de alcohol, estaría combatiendo en primera línea las zonas de menor a mayor carga viral o bacteriana.

2. FUNDAMENTACIÓN TEÓRICO

Brote epidémico: Definición:

Es una clasificación utilizada en epidemiología para referirse al brote de una enfermedad infecciosa en un lugar y momento determinado.[41]

Diferencia entre epidemia, pandemia y endemia:

La epidemia se clasifica como una enfermedad activa y se está extendiendo rápidamente con un aumento significativo en el número de casos, aunque todavía dentro de un área geográfica específica.[41]

Para declarar una pandemia se deben cumplir dos criterios: la enfermedad afecta a muchos continentes, y los casos en cada país ya no son importados sino trasladados a la comunidad. La enfermedad endémica se define como la presencia persistente de una enfermedad en un área geográfica o población, aunque también puede referirse a una alta prevalencia de enfermedad crónica en un área o población determinada. Para ello, se deben cumplir dos criterios simultáneamente: la enfermedad persiste en el tiempo y atraviesa un territorio o población bien definidos.[41]

Coronavirus o Covid-19:

La enfermedad por coronavirus (COVID-19) es una enfermedad infecciosa causada por el virus SARS-CoV-2. La mayoría de las personas infectadas con el virus tienen una enfermedad respiratoria de leve a moderada que desaparece por sí sola sin un tratamiento específico. Sin embargo, algunos estarán gravemente enfermos y necesitarán atención médica. Los ancianos y las personas con enfermedades básicas como la enfermedad cardiovascular, la diabetes, la enfermedad respiratoria crónica o el cáncer y las enfermedades graves se desarrollan. Cualquier persona, independientemente de su edad, puede infectarse con COVID-19 y enfermarse gravemente o morir.[23]

La mejor manera de prevenir y retrasar la transmisión es estar bien informado sobre la

enfermedad y cómo se propaga. Protéjase a sí mismo y a los demás de las infecciones manteniéndose al menos a un metro de distancia de los demás, usando una máscara adecuada y lavándose o desinfectándose las manos con frecuencia con un desinfectante para manos a base de alcohol. Vacúnese cuando sea su turno y obedezca las leyes locales. El virus se puede propagar a través de la boca o la nariz de una persona infectada en forma de pequeñas partículas líquidas cuando tose, estornuda, habla, canta o respira. Estas partículas van desde las gotas respiratorias más grandes hasta los aerosoles más pequeños. Es importante practicar buenas prácticas de respiración, como toser en los codos, doblar y en casa y aislar hasta que recupere la recuperación si no se siente bien.[23].

La OMS caracteriza a Covid-19 como pandemia:

La enfermedad por coronavirus (COVID-19) es una enfermedad infecciosa causada por el virus SARS-CoV-2. La mayoría de las personas infectadas con el virus tienen una enfermedad respiratoria de leve a moderada que desaparece por sí sola sin un tratamiento específico. Sin embargo, algunos estarán gravemente enfermos y necesitarán atención médica. Los ancianos y las personas con enfermedades básicas como la enfermedad cardiovascular, la diabetes, la enfermedad respiratoria crónica o el cáncer y las enfermedades graves se desarrollan. Cualquier persona, independientemente de su edad, puede infectarse con COVID-19 y enfermarse gravemente o morir.[24]

La mejor manera de prevenir y retrasar la transmisión es estar bien informado sobre la enfermedad y cómo se propaga. Protéjase a sí mismo y a los demás de las infecciones manteniéndose al menos a un metro de distancia de los demás, usando una máscara adecuada y lavándose o desinfectándose las manos con frecuencia con un desinfectante para manos a base de alcohol. Vacúnese cuando sea su turno y obedezca las leyes locales. El virus se puede propagar a través de la boca o la nariz de una persona infectada en forma de pequeñas partículas líquidas cuando tose, estornuda, habla, canta o respira. Estas partículas van desde las gotas respiratorias más grandes hasta los aerosoles más pequeños. Es importante practicar buenas prácticas de respiración, como toser en los codos, doblar y en casa y aislar hasta que recupere la recuperación si no se siente bien.[24]

Tabla 1. Variación en casos por reclasificación Ecuador- Covid 1- 2020 [25]

Provincia	07/05/2020		08/05/2020	
	Casos		Casos	
	Confirmados	Descartados	Confirmados	Descartados
AZUAY	561	1859	565	1862
BOLÍVAR	197	159	159	116
CAÑAR	349	758	239	733
CARCHI	84	124	92	125
CHIMBORAZO	229	162	223	154
COTOPAXI	148	167	156	157
EL ORO	673	1231	652	1285
ESMERALDAS	290	316	283	309
GALÁPAGOS	70	768	70	768
GUAYAS	12577	92052	11532	7057
IMBABURA	108	240	114	239
LOJA	274	605	252	622
LOS RÍOS	1049	877	966	614
MANABÍ	1263	831	1173	685
MORONA SANTIAGO	71	195	40	199
NAPO	40	70	43	70
ORELLANA	48	35	52	35
PASTAZA	80	101	82	102
PICHINCHA	1956	4088	1988	4187
SANTA ELENA	608	209	481	144
SANTO DOMINGO DE LOS TSÁCHILAS	332	490	353	498
SUCUMBÍOS	42	175	52	437
TUNGURAHUA	234	223	248	222
ZAMORA CHINCHIPE	78	124	66	129
Total	21361	23012	19881	20749

2.1. ANTECEDENTES

El proyecto de titulación que se está desarrollando es un robot móvil diferencial aspersor de alcohol, capaz de movilizarse por telemetría, será utilizado exclusivamente para el uso de fumigación en zonas de precaución o riesgos lo cual será operado por cualquier dispositivo vinculado a internet. A continuación, vamos a conocer todos sus componentes:

2.2. ROBOT MÓVIL

Los robots móviles, también conocidos como AMR (acrónimo de Autonomous Mobile Robots en inglés), son un conjunto de sistemas diseñados para realizar tareas específicas de forma sistemática y repetitiva.

Los dispositivos de este tipo tienen ciertas características, como la capacidad de moverse mediante software, sensores de movimiento, métodos determinados de acuerdo con ciertas tecnologías (por ejemplo, inteligencia artificial, aprendizaje automático, etc.).**¡Error! No se encuentra el origen de la referencia.**

2.2.1. ROBOT MÓVIL CON PLATAFORMA DE RUEDAS DIFERENCIAL

El diseño diferencial es uno de los sistemas de movimiento menos complicados. El robot puede moverse en línea recta, rotar y deslizarse a la misma velocidad. Cuando los motores encuentran diferentes resistencias (una rueda en la placa y la otra en la muela abrasiva), las velocidades de los motores serán diferentes y el robot girará incluso si inicialmente está configurado para ir en línea recta. Esto se debe a que la velocidad debe controlarse dinámicamente, es decir, debe haber una forma de controlar y cambiar la velocidad del motor a medida que el robot avanza.**¡Error! No se encuentra el origen de la referencia.**

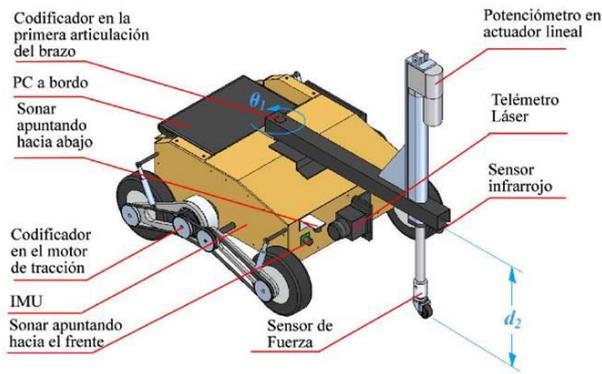


Figura 2. Estructura de un robot diferencial ¡Error! No se encuentra el origen de la referencia.

2.3. RASPBERRI PI

Es una pequeña computadora económica que es lo suficientemente pequeña como para caber en la palma de su mano, pero se conecta a su televisor y teclado y funciona como cualquier otra computadora. ¡Error! No se encuentra el origen de la referencia.



Figura 3. Ordenador Raspberry. ¡Error! No se encuentra el origen de la referencia.

2.4. ARDUINO

Arduino es una plataforma de programación basada en una placa electrónica de hardware libre que integra un microcontrolador programable y múltiples buses. Ayudan a establecer la comunicación entre el microcontrolador y varios sensores y actuadores.[4]



Figura 4. Microprocesador Arduino Mega. [4]

2.5. NODE RED

Es una herramienta de visualización basada en secuencias desarrollada originalmente por IBM para conectar dispositivos de hardware, API y servicios en línea en Internet de las cosas. Expone dependencias y funciones que permiten al usuario programar sin escribir una sola línea de código. El botón rojo es un editor de flujo basado en navegador donde puede agregar o eliminar nodos y conectarlos para que interactúen entre sí. ¡Error! No se encuentra el origen de la referencia.

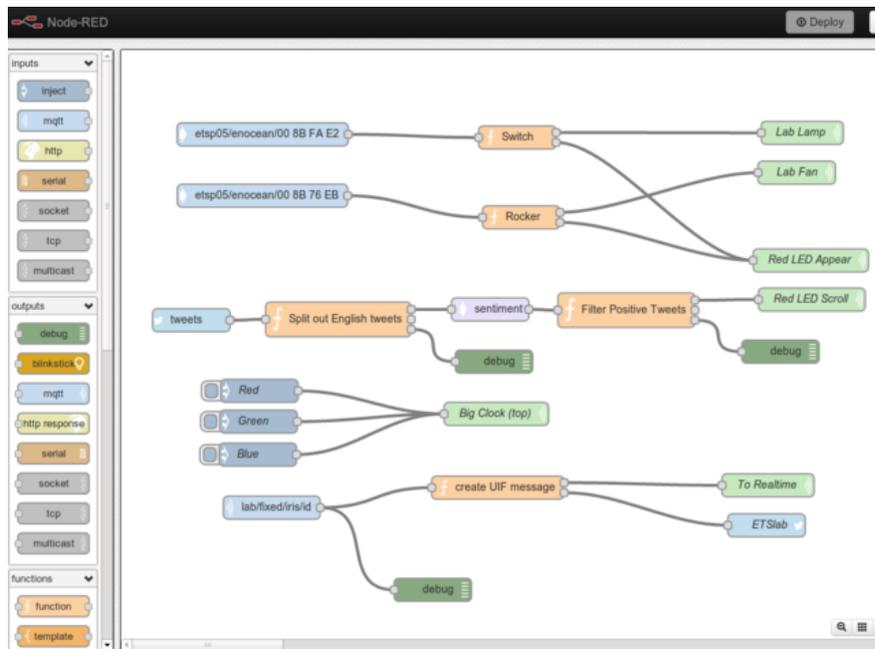


Figura 5. Esquema de programación Node red. [5]

2.6. IBM CLOUD

La plataforma es escalable y es compatible con pequeños equipos y organizaciones de desarrollo, así como con grandes empresas. Implementada en centros de datos de todo el mundo, la solución se inicia de manera rápida, confiable y opera en un entorno probado y de apoyo en el que puede confiar.[6]

Ofrece soluciones que brinden niveles más altos de cumplimiento, seguridad y capacidad de administración con patrones arquitectónicos probados y métodos de entrega flexibles para respaldar cargas de trabajo de misión crítica y entregar Brindar la nube pública más segura y abierta para empresas con plataformas de nube híbrida de próxima generación, de vanguardia capacidades de datos e inteligencia artificial, y una profunda experiencia comercial en 20 industrias. Existen soluciones que se adaptan a sus necesidades para la nube pública, local o híbrida. **¡Error! No se encuentra el origen de la referencia.**

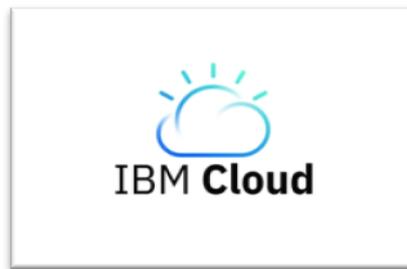


Figura 6. Logotipo IBM Cloud [6]

2.7. IoT

En general, el Internet de los artículos incluye todo lo que se puede conectar a Internet, desde equipos y vehículos en fábricas hasta dispositivos móviles y horas intelectuales. Pero hoy, IoT menciona más específicamente los componentes conectados equipados con sensores, software y otras tecnologías que les permiten enviar y recibir datos. Tradicionalmente, la conectividad se ha brindado principalmente a través de Wi-Fi, mientras que hoy en día 5G y otros tipos de plataformas de red son cada vez más capaces de manejar big data con velocidad y confiabilidad. **¡Error! No se encuentra el origen de la referencia.**



Figura 7. Conectividad por IoT [7]

2.8. CLIENTE/SERVIDOR

La expresión cliente-servidor se utiliza en el campo de la informática. En este contexto, un cliente es un dispositivo que solicita ciertos servicios del servidor. A su vez, el término servidor se refiere a un dispositivo que brinda servicios a las computadoras (computadoras) conectadas a él a través de una red. Así, el término cliente-servidor o cliente-servidor se refiere a un modelo de comunicación que conecta diferentes dispositivos informáticos a través de una red. En este contexto, el cliente envía solicitudes de servicio al servidor y el servidor es responsable de responder a estas solicitudes.[8]

.

2.9. DASHBOARD

Es una herramienta de administración de información que rastrea, analiza y muestra indicadores clave de rendimiento (KPI), métricas y datos esenciales para monitorear el estado de su negocio, departamento, campaña o proceso en particular.[9]

Podemos pensar en los dashboard como una especie de “resumen” que recopila datos de diferentes fuentes en un solo lugar y los presenta de una manera fácil de entender para que lo más importante llegue al usuario. **¡Error! No se encuentra el origen de la referencia.**

2.10. DESINFECCIÓN

La desinfección es una operación para destruir o destruir microorganismos y/o inactivar virus no deseados. La desinfección se agrupa en 5 actividades diferentes:[10]

- Los productos bactericidas matan las bacterias[10]
- Candidiasis[10]
- Fungicidas Un producto que mata hongos (esporas y mohos)[10]
- Esporicida Un producto que mata las esporas bacterianas.[10]
- Virucide producto que mata virus[10]

Los resultados de la operación de esterilización se limitan a los microorganismos presentes durante la operación. La desinfección no previene una nueva infección y, por lo tanto, debe repetirse con frecuencia. Desinfección - una actividad con un efecto temporal.[10]

2.11. TELEMETRÍA

Es una tecnología que permite la medición y control remoto de variables físicas o químicas a partir de datos enviados a un centro de control. La transmisión de información al operador en un sistema de telemetría se suele realizar mediante comunicación inalámbrica, aunque también se puede realizar por otros medios (teléfono, red informática, comunicación, etc., conexión de fibra óptica, etc.). Los sistemas de telemetría reciben las instrucciones y datos que necesitan para operar desde el Centro de Control o la plataforma de navegación web. **¡Error! No se encuentra el origen de la referencia.**

2.12. SENSOR ULTRASÓNICO

Es un sensor de distancia económico, su uso es muy común en la industria de la robótica, utiliza un transductor ultrasónico para detectar objetos. Funciona emitiendo un sonido ultrasónico a través de uno de sus transductores, mientras espera el sonido del objeto presente, el sonido será captado por el otro transductor. La distancia es proporcional al tiempo de ocurrencia del eco. **¡Error! No se encuentra el origen de la**

referencia.



Figura 8. Sensor ultrasónico de Arduino [12]

2.13. DRIVER

El controlador es un transistor intermedio que mantiene la corriente requerida para impulsar el transistor de potencia. Los preamplificadores son transistores muy pequeños, de alta ganancia y estables para amplificar señales muy débiles, pero no pueden controlar los transistores de la etapa de potencia. Bueno, el controlador es ese transistor intermedio, que hace el trabajo de suministrar corriente a la etapa de potencia que los anteriores no eran capaces de hacer. Esto se llama configuración en cascada, donde un preamplificador aumenta el voltaje de las señales muy débiles hasta que son manejables. Estas señales se aplican a un controlador que las amplifica (actualmente) lo suficiente como para enviarlas a la etapa de potencia, que es el paso final necesario para entregarlas a la carga que las consume (altavoces, motor, etc.). Cada etapa puede constar de varias etapas, es decir, un preamplificador puede constar de dos etapas amplificadoras, un controlador de una etapa y una etapa de potencia de dos.[13]

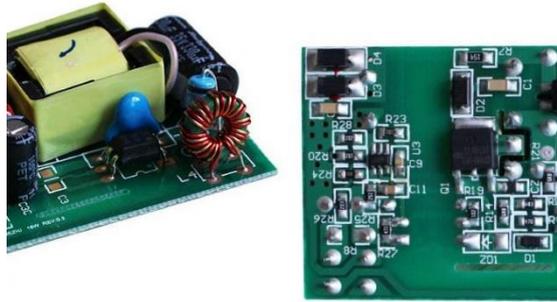


Figura 9. Tarjeta driver para led de tensión constante o corriente constante.

¡Error! No se encuentra el origen de la referencia.

2.14. CÁMARA IP

Una cámara IP consta principalmente de una lente, un sensor de imagen, un procesador de imagen, un SoC para la compresión de video (System on chip) y un chip Ethernet que proporciona conectividad de red para la transmisión de datos.

Cuando la luz pasa a través de la lente hacia el sensor, se convierte en una señal digital y luego es procesada por el procesador de señal digital incorporado. Los datos de video procesados luego son comprimidos por los medios SoC para reducir el tamaño de los datos para una transmisión óptima. Finalmente, la imagen de video se envía a través de la red como un flujo de video a los terminales para su visualización y almacenamiento. En este artículo, nos centraremos en el paso final del proceso, pasar información al sistema de administración/registro/visualización.[26]



Figura 10. Cámara IP [26]

2.15. SENSOR DE NIVEL TIPO BOYA

El sensor de nivel, también conocido como "interruptor de nivel" o "sensor de flotador", es un dispositivo que funciona en combinación con un interruptor de contacto (interruptor de trapeador) y un flotador magnético. Flotador de movimiento para abrir y cerrar contactos eléctricos.[38]

Detecta el nivel de líquido en los depósitos y depósitos en los que está instalado, señalando con una señal ON/OFF que se ha alcanzado el nivel de lleno, vacío u otro especificado en diseño. Debido a que están fijos en un lugar del tanque, los sensores de nivel no se ven afectados por las olas o las vibraciones, lo que proporciona una mayor confiabilidad y repetibilidad que otros tipos de sensores de nivel, como los flotadores de nivel antiguos.[38]



Figura 11 Sensor de nivel tipo boya [38]

2.16. CARGA BACTERIANA

Es el número y tipo de microorganismos viables presentes en el elemento. En pocas palabras, el crecimiento de bacterias es exponencial. Una célula bacteriana con un tiempo de generación de 18 a 20 minutos durante 5 a 8 días producirá una masa de población que excede la masa de la Tierra. Este sesgo teórico se ve amenazado por los nutrientes, la acidificación metabólica, el antagonismo del microbioma preexistente y las frecuentes mutaciones incompatibles. Cualquier patógeno debe adaptarse y reproducirse solo en circunstancias favorables. Pero lo harán "en silencio" para no

provocar la hostilidad de quienes los rodean. Cuando un grupo de bacterias ven buenas condiciones. [14]

2.17. NODE.JS (JASON)

Node.js es el tiempo de ejecución de JavaScript (de ahí su sufijo .js, que hace referencia al lenguaje JavaScript). Este tiempo de ejecución en tiempo real incluye todo lo que necesita para ejecutar un programa de JavaScript. Utiliza un modelo de E/S controlado por eventos sin bloqueo, lo que lo hace liviano y eficiente (la entrada es una solicitud y la salida es una respuesta). Esto puede referirse a cualquier operación, desde leer o escribir un archivo de cualquier tipo hasta realizar una solicitud HTTP. está completamente dirigido por eventos. En pocas palabras, el servidor consta de un hilo que procesa un evento tras otro. El tipo de evento se genera en una nueva solicitud. El servidor comienza a procesarlo, y cuando ocurre una operación de bloqueo de E/S, no esperará a que se complete la operación, sino que generará una función de devolución de llamada. El servidor inmediatamente comienza a manejar otro evento (posiblemente otra solicitud) y una vez que se completa la E/S, continúa trabajando en la solicitud llamando nuevamente cada vez que tiene tiempo para hacerlo.[15]

2.18. ARCHIVO.JS

JASON, abreviatura de JavaScript Object Notation o JavaScript Object Notation, es un formato de intercambio de datos liviano que es fácil de leer y escribir para los programadores, y fácil de interpretar y generar para las máquinas.[16]

2.19. INDUSTRIA 4.0

Industria 4.0 se refiere a una nueva fase de la revolución industrial que se centra en gran medida en la interoperabilidad, la automatización, el aprendizaje automático y los datos en tiempo real, también conocidos como IoT o fabricación integrada e inteligente. Integra operaciones físicas y de fabricación con tecnología digital inteligente, aprendizaje automático y grandes datos. Creando un ecosistema más cohesionado y mejor conectado para las empresas que fabrican y gestionan

productos.[17]

2.20. CLOUD IBM VS OTROS CLOUD

IBM vs AWS

AWS ofrece solo ofrece 6 servicios dedicados o bare metal de configuración a diferencia de IBM que ofrece cientos. [18]

IBM vs Google Cloud

Google no puede ofrecer servidores bare metal y virtuales. [18]

IBM vs Alibaba

IBM ofrece características, servicios y productos que Alibaba no ofrece. [18]

2.21. BARE METAL

Demuestra que este producto pertenece a un concepto basado en la parte física del almacenamiento, que es el hardware. De hecho, un servidor bare-metal no es más que lo que alguna vez se llamó un "servidor dedicado", una computadora en el centro de datos de un proveedor de alojamiento con recursos fácilmente disponibles y para un solo cliente. En este sentido, los servidores bare metal se diferencian de los productos tradicionales de alojamiento compartido, como los servidores virtuales, donde los proyectos web de diferentes clientes se alojan en la misma plataforma de hardware en la misma plataforma y máquinas virtuales separadas.[19]

2.22. MICROSOFT AZURE

Incluye más de 200 productos y servicios en la nube diseñados para ayudarlo a proporcionar vida a las nuevas soluciones que le permiten resolver las dificultades actuales y crear un futuro. Crear, implementar y controlar aplicaciones en algunas nubes, en un entorno local y alrededor del circuito, utilizando herramientas y empleados preferidos. Implemente máquinas virtuales Windows y Linux, modernice sus aplicaciones y amplíe más en entornos híbridos y de nube.[27]

2.23. OBS STUDIO

Esta es una aplicación multiplataforma gratuita que le permite capturar o grabar videos y retransmitirlos. La aplicación es profesional y fácil de usar. Las características de OBS son aquellas que se pueden especificar en cualquier situación donde se requiera grabación, grabación de video o retransmisión, con énfasis en:

- Grabación y mezcla de video en tiempo real. • Permite incluir múltiples fuentes: ventanas, imágenes, texto, webcams, navegadores, pestañas de captura.[28]
- Configurar un número ilimitado de escenas con la posibilidad de introducir cambios mediante transiciones. [28]
- Una herramienta para mezclar audio con filtros.[28]
- Admite transmisión de video a las plataformas de video más populares o utilizando cualquier formato de transmisión de red estándar. [28]
- Ampliar su negocio complementando y complementando tanto los existentes como desarrollando otros nuevos.[28]

2.24. ROBOTS FUMIGADORES EN OTROS PAÍSES

Robot Trakür

Investigadores del INTA Castelar, en Buenos Aires, Argentina. Diseñaron un sistema de cobertura de cultivo cerrado para proteger la salud del operador.

Es un vehículo autopropulsado, potente y económico que viaja en un sistema de guías o rieles virtuales: un cable que emite señales y es recogido por el robot a partir de sensores. Funciona con energía eléctrica y se alimenta de una batería de 12V que le permite viajar a velocidades de entre 1.4 a 4.2 kilómetros por hora. Además, cuenta con una autonomía de más de ocho horas y una cámara a través de la cual monitoriza el dispositivo de forma remota.[20]

Prototipo de control de plagas con sistema de rociadores completo y suficiente batería para completar un turno. Tiene un sistema de transmisión inalámbrica que lo monitorea de forma remota. [20]

Robot móvil Aurora.

En España, desarrollaron el robot de invernadero móvil AURORA. Ha sido desarrollado por la Facultad de Ingeniería y Sistemas Automatizados de la Universidad de Málaga y el Grupo de Control y Visión Robótica de la Universidad de Sevilla, específicamente para tareas de esterilización. Incorpora una mecanización basada en un motor AC, el cual es alimentado por un generador de 220V AC con una capacidad de 2,5KW. Su sistema de control está basado en un PC industrial. Su sistema de detección se basa en sensores ultrasónicos analógicos y digitales que se pueden utilizar en el control reactivo. Incorpora una cámara CCD para control remoto. [20]

Robots agrícolas desarrollados por la universidad de Illinois

En los Estados Unidos, ingenieros agrícolas de la Universidad de Illinois han desarrollado una serie de robots pequeños y económicos (\$150 a \$500 cada uno) diseñados específicamente para realizar tareas agrícolas y reemplazar máquinas. Actualmente se utilizan ganchos pesados y costosos para plantar, fumigar, cosechar y arado [20]

Robots de fumigación de invernaderos (ESIME)

En el contexto mexicano encontramos una tesis de ESIME, que es un robot móvil para la fumigación de invernaderos. Es un robot móvil autónomo con movimiento de ruedas de tipo diferencial, el movimiento está asegurado por dos motores DC alimentados por baterías recargables para la autonomía, es guiado por sensores y controlado por una máquina de estados. [20]

3. MARCO METODOLÓGICO

3.1. DESCRIPCIÓN DEL PROTOTIPO

Este trabajo se desarrolló utilizando tecnología de fabricación y servicio de Industria 4.0 controlado por un nodo de red, se trata de un prototipo de robot cuyo cerebro de una mini computadora es una tarjeta Raspberry que permite integrar software a otros dispositivos, comandos virtuales y conexión wifi.

El software para el control y procesos de fumigación cuenta con una interfaz gráfica (Dashboard) amigable al usuario la cual también esta enlazados con la nube (Cloud) y una cámara IP, esto a su vez estará comunicado con un microprocesador (Arduino) que comunicara a los demás componentes, los procesos que se desarrollan el prototipo estarán detalladas paso a paso para una mejor comprensión del alumnado o docente

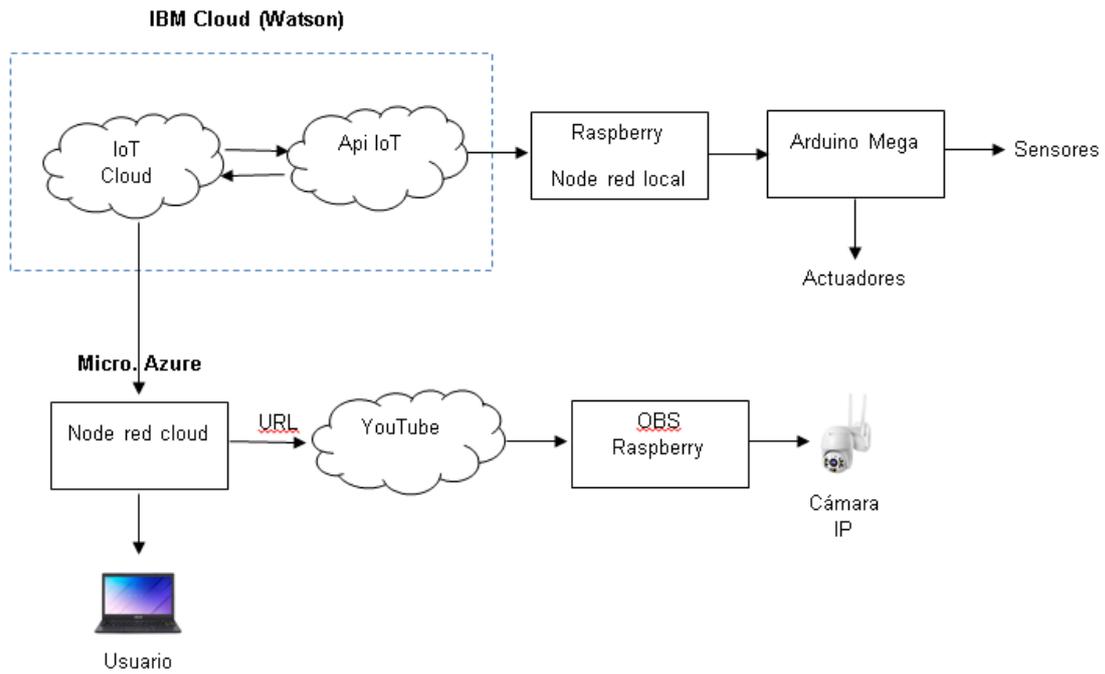


Figura 12 Diagrama de bloques del robot aspersor

3.2. ETAPAS DEL PROTOTIPO

Para el desarrollo del prototipo se aplicaron 3 etapas, la etapa del software o procesamiento, segunda la etapa de adquisición y como tercera la etapa de aplicación.[39]

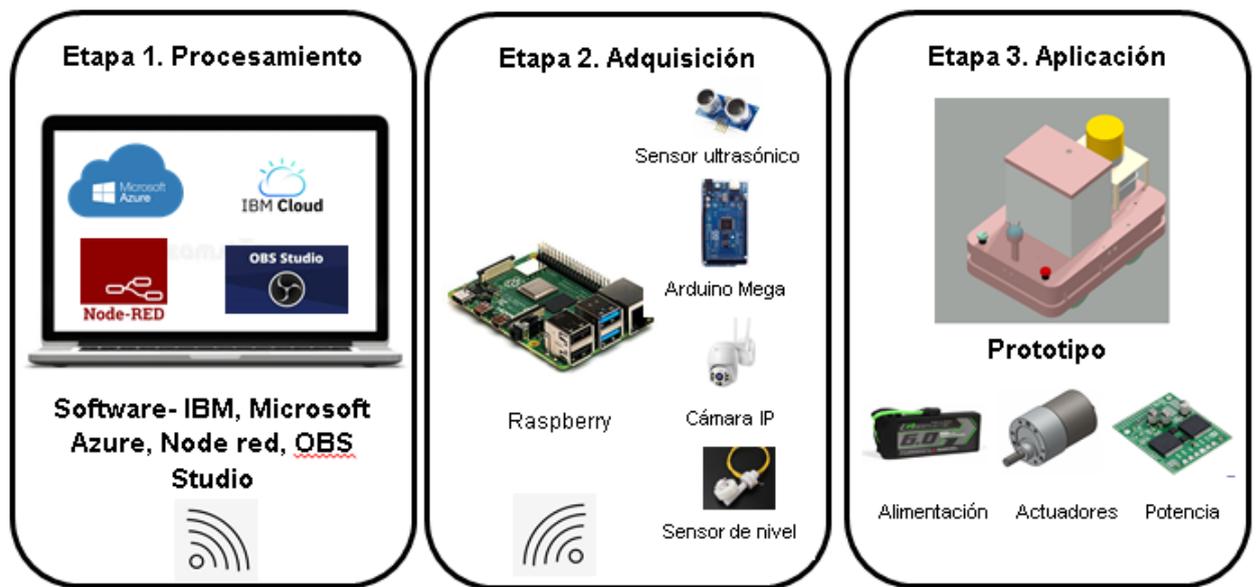


Figura 13. Etapas del prototipo

3.3. ETAPA DE PROCESAMIENTO

En el procesamiento se realizó las rutinas de las prácticas a realizar en el prototipo, para el proceso de indicaciones de movimiento y fumigación se ejecuta con el software Node red a la par con Microsoft Azure e IBM Cloud para crear una API, que este a su vez en la nube con un tablero de comandos (dashboard), todo este proceso se crea con procesamiento de datos Microsoft Azure y al mismo tiempo con servicios de IBM Cloud como son IBM Watson, la cual hacen de enlace para la comunicación del Node red tanto de local como en la nube; la cámara posee una conexión mediante wifi la cual se conecta a Cloud Cam la cual recibe esos datos OBS Studio y por medio de Youtube se genera un URL para enlazarlo al Node red. En el siguiente diagrama de flujo (figura 14) se muestran los pasos de la ejecución hacia la etapa 2.

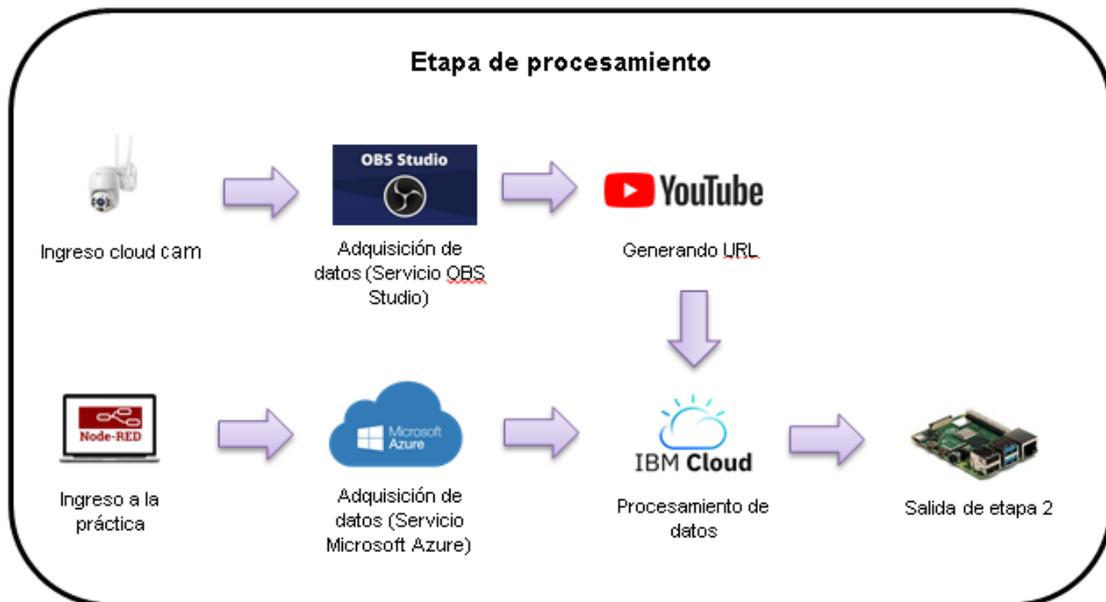


Figura 14. Etapa de procesamiento

3.4. ETAPA DE ADQUISICIÓN

En esta etapa se encarga de verificar los estados de lecturas de los diferentes sensores de distancia y nivel conectados al microprocesador Arduino que a su vez estará conectado por comunicación serial al Raspberri para enviar a la etapa de procesamiento, con el retorno se encargará de transmitir los datos de control a los actuadores

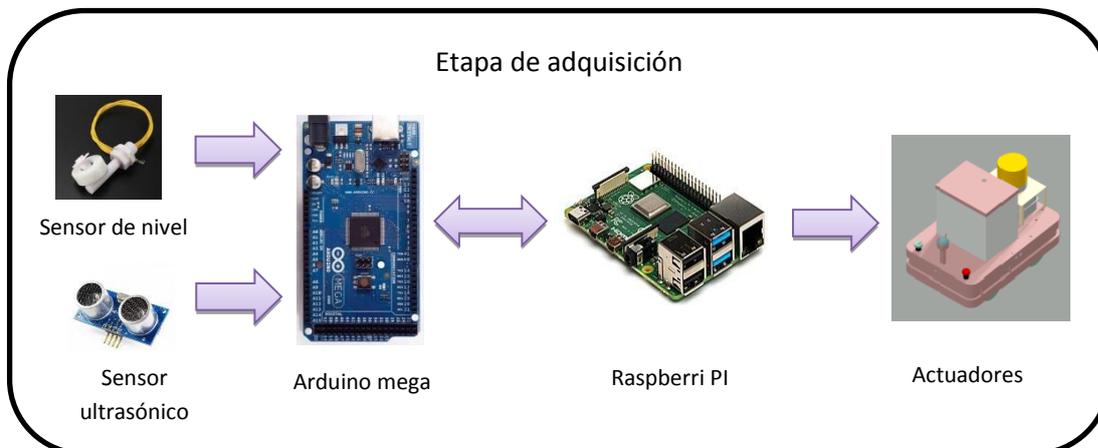


Figura 15. Etapa de adquisición

3.5. ETAPA DE APLICACIÓN

En esta última etapa es el prototipo, son los elementos que logran que el robot tenga movimiento tal como motores, estructuras, drivers, luminarias e interruptores, en la figura siguiente se muestran los elementos del robot.

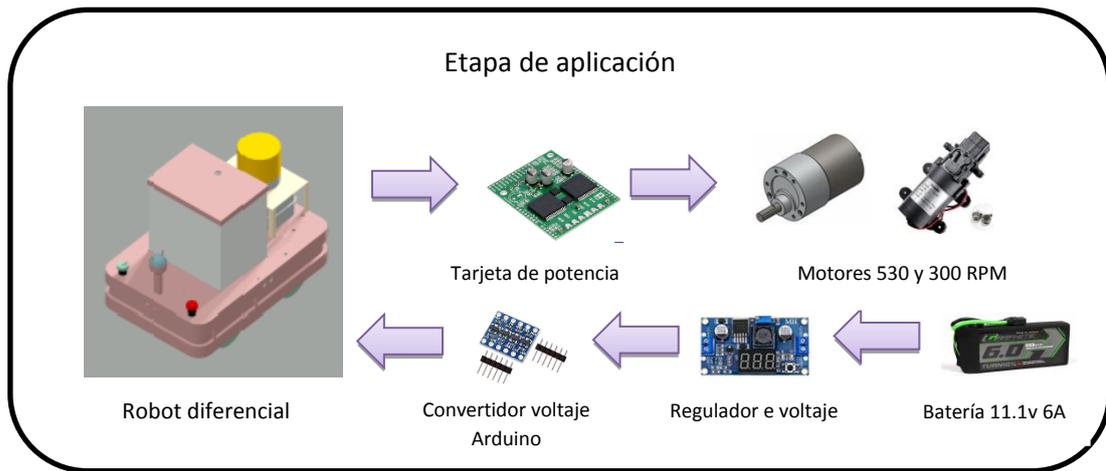


Figura 16. Etapa de aplicación

3.6. DIAGRAMA DE CONEXIONES

El prototipo posee conexiones con los diferentes periféricos como sensores, actuadores (motores de bomba y agua), tarjeta de potencia, tarjeta convertidora, tarjeta reguladora de voltaje, cámaras que se describen en esta sección.

3.7. CONEXIÓN TARJETA DE POTENCIA

La conexión de la tarjeta de potencia para el control de los motores en mediante 3 tarjetas la cual 2 manejan los motores de tracción y la otra el motor de fumigación la cual la conexión será mediante los pines del Arduino como se muestra en la tabla.

Tabla 2. Entradas de la tarjeta de potencia

ENTRADAS DE LAS TARJETA DE POTENCIA									
	GND	M1E N	M1IN A	M1IN B	M1P WM	M2E N	M2IN A	M2IN B	M2P WM
PLACA 1	GND	6	5	4	10	2	14	15	9
PLACA 2	GND	40	38	36	12	34	32	30	11
PLACA 3	GND	42	44	46	8				

Tabla 3. Salidas de la tarjeta de potencia

SALIDAS DE TARJETAS DE POTENCIA						
	M1A	M1B	GND	VIN	M2A	M2B
PLACA1	+M1	-M1	-BATE	+BATE	+M2	-M2
PLACA 2	+M3	-M3	BATE	+BATE	+M4	-M4
PLACA 3	+BOM	-BOM	-BATE	+BAT		

3.8. CREACIÓN DE CUENTA IBM

Para crear una cuenta en IBM Cloud gratuita se procede como primer paso a entrar en el link **cloud.ibm.com** la cual sirve para adquirir el servicio de internet de las cosas (internet of thing) de forma remota y también el servicio de Node red en la nube.

Se ingresa al link ya mencionado y procedemos a ingresar un correo, una clave de seguridad y datos personales tal cual como crear un correo personal.

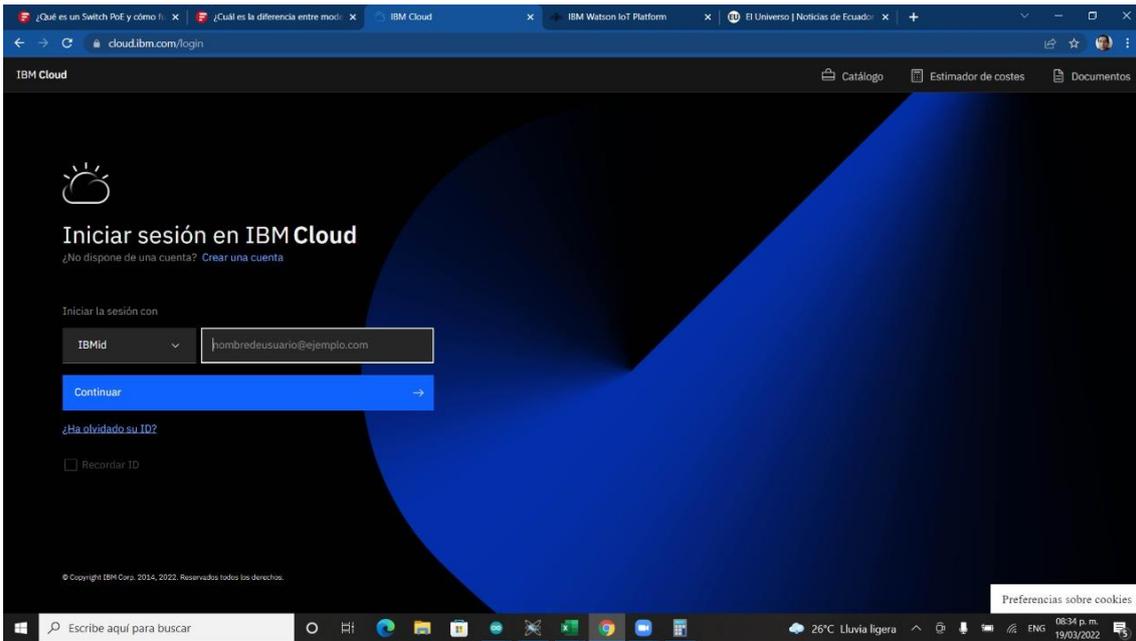


Figura 17 Ingreso a IBM Cloud para creación de cuenta

Se ingresa a la lista de recursos nos dirigimos en servicios y software, y se procede a dar click en la opción internet de las cosas-IC (internet of thing-IC).

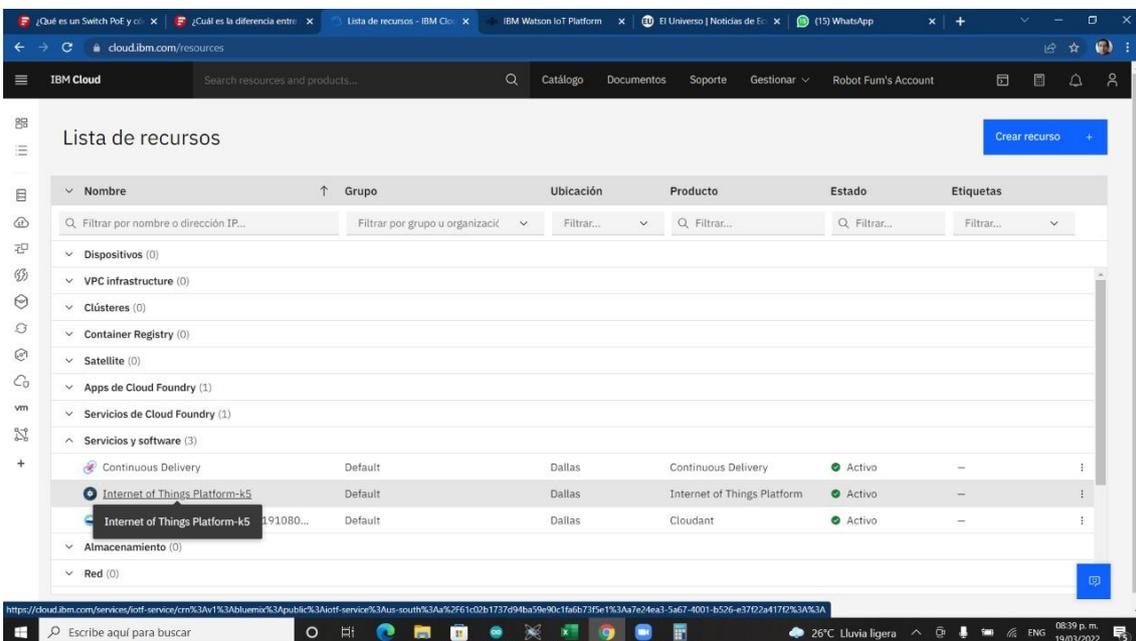


Figura 18. Creando servicio IoT

3.9. IBM WATSON IOT

Una vez ingresado se procederá a crear un servicio para controlar algunos dispositivos por medio de la plataforma Watson IoT la cual se seleccionará la opción lanzar.



Figura 19. Creando el servicio Watson IoT

Luego se ingresa al servicio de Watson IoT en la cual nos va a permitir agregar y ver el dispositivo que se desea: se procede a escoger agregar dispositivo la cual se encuentra en superior derecha de la ventana.

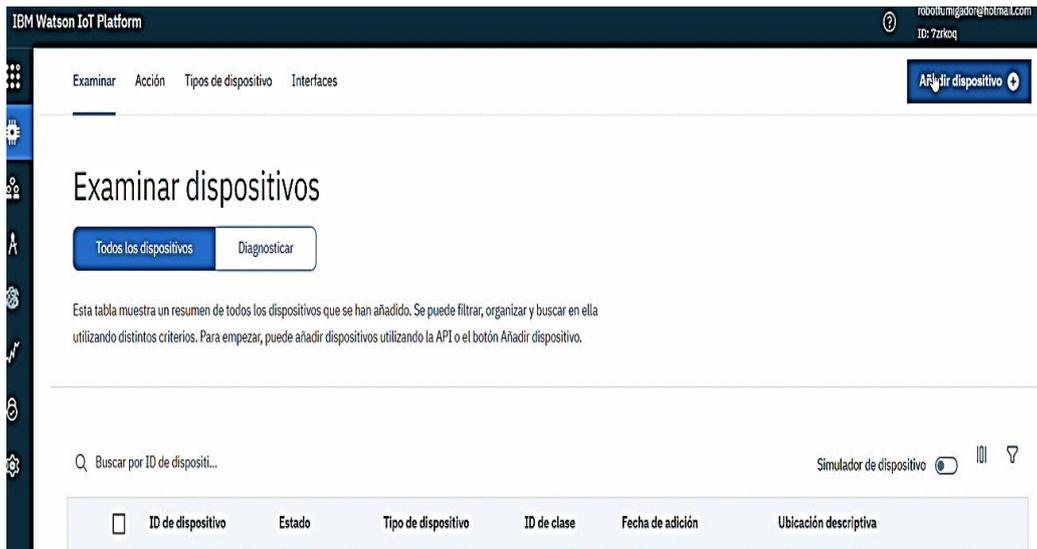


Figura 20. Agregando un dispositivo a IBM Watson

Hecho el paso anterior se abrirá una ventana para escoger qué tipo de dispositivo se desea ejemplo: Arduino, interruptores inteligentes, televisores, controladores lógicos, etc.; la cual también se podrá poner una ID o nombre; una vez ingresado esos datos se da click en siguiente y avanzar.



Figura 21. Identificando al nuevo dispositivo IoT

En la siguiente ventana se abrirá la configuración de seguridad de nuestro dispositivo, se tiene la opción de poner una clave correspondiente a la señal, en este caso se accede a poner una seguridad para que la señal no sea de uso público sino privada, para evitar que el servicio no se caiga de nuestra plataforma. En esta ventana,

además brindará un resumen de la señal creada y de lo que se ha realizado. Seguido damos click en finalizar.

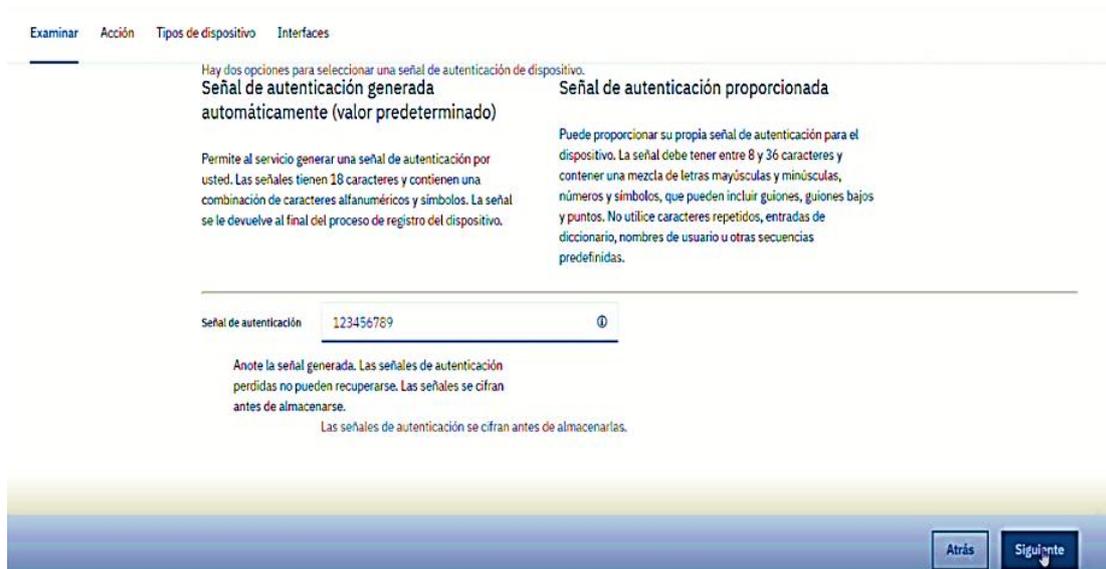


Figura 22. Dando seguridad a la señal IoT

Finalizado se escoge la opción volver o atrás y se podrá visualizar el servicio que se ha adquirido. Agregando el dispositivo IOT para esta plataforma, lo que se hace es una comunicación bidireccional esto quiere decir enviar del IOT hacia un Node red local, para visualizar nos dirigimos a panel – aplicaciones.



Figura 23. Generando comunicación bidireccional de IoT a node red

En la ventana de aplicaciones se escoge la opción generar clave API. En la siguiente ventana nos dará la información de la descripción de nuestra clave y la caducidad, en este caso no queremos que haya caducidad así que dejamos en blanco y damos siguiente.

Generar clave de API

Información Permisos

Descripción: Hola

Caducidad de la clave de: Desactivado Activado

API: Seleccionar fecha

Cancelar Siguiente

Figura 24. Generando información a la clave API

En la siguiente ventana se obtiene los permisos de la clave API, de la aplicación creada como se explica en la Figura 16. Se procede a dar un rol de acceso, en este caso se procede a escoger la aplicación de dispositivos y se procede a dar click en generar clave, es importante recordar nuestra clave y nuestra señal de autenticación ya que si se olvida no se podrá recuperar todo lo hecho. Como siguiente se selecciona la opción cerrar la cual nos enviara un mensaje de aviso de la clave .

Generar clave de API

Información Permisos

La aplicación tendrá acceso para el rol siguiente:

Rol: Aplicación de visualización

Para obtener más información: [usuario, aplicación y pasarela.](#)

Aplicación de confianza en segundo plano

Aplicación de procesador de datos

Aplicación de dispositivo

Aplicación de operaciones

Aplicación estándar

Aplicación de visualización

Atrás Generar clave

Figura 25. Generando un rol de acceso a la API

Creado estas dos herramientas serán necesarias (IOT Watson y rol de API), para que así, comunique desde Node red hasta IBM Watson de IOT y para comunicarnos de forma local; IOT plataforma se conecta por medio de nuestra clave API y hace puente para comunicarse de forma local.

Teniendo todo en orden se procede a dirigir a la lista de recurso de nuestro IBM Cloud que esta superior izquierda. Buscamos la opción fundición en la nube (cloud foundry), si nuestra app esta enlazada correctamente se selecciona y se va a obtener la dirección IP de la API o en su defecto en la parte superior del centro de la ventana damos click en visitar URL de APP.

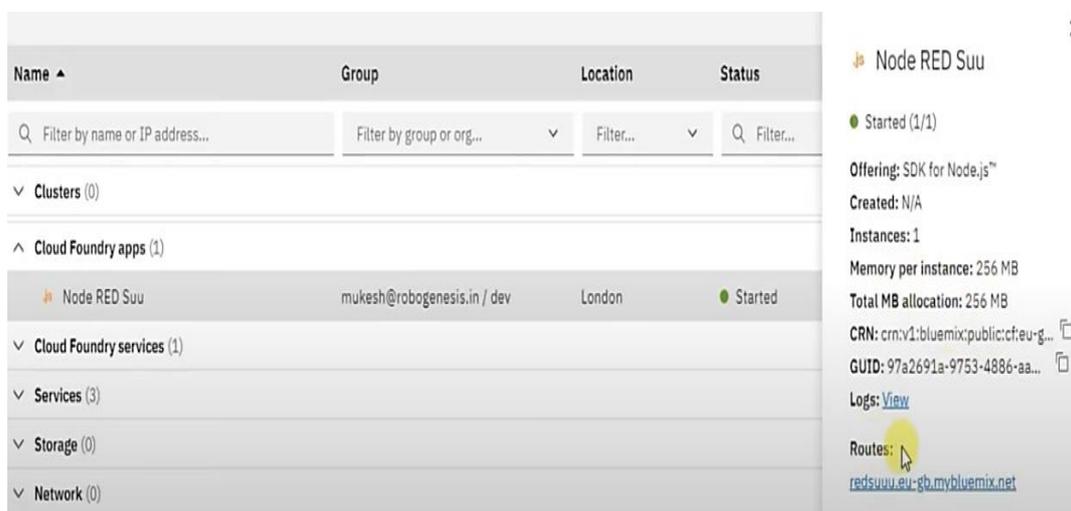


Figura 26. Lista de recursos de IBM Cloud.

3.10. MICROSOFT AZURE

Se procede a crear una cuenta en Microsoft Azure la cual se buscara en Google; una vez ingresado al link seleccionamos en la opción empieza gratis.



Figura 27. Ingresando a Microsoft Azure

Hecho esto se ingresa el usuario y contraseña de una cuenta Outlook, luego se debe escoger el método de pago y el ancho de banda que deseamos que eso ya se selecciona dependiendo al usuario.



Figura 28. Seleccionando método de pago para Azure

Se procede a crear nuestra máquina virtual seleccionándola en los servicios de Azure que se encuentra en la parte superior de la ventana

Servicios de Azure



Figura 29. Creando máquina virtual

Ingresado se va a abrir una ventana de las máquinas virtuales creadas, se selecciona: agregar >máquina virtual la cual se abra una nueva ventana para modificar los datos de dicha opción.

Crear una máquina virtual ...

Datos básicos Discos Redes Administración Opciones avanzadas Etiquetas Revisar y crear

Cree una máquina virtual que ejecuta Linux o Windows. Seleccione una imagen de Azure Marketplace o use una imagen personalizada propia. Complete la pestaña Conceptos básicos y, después, use Revisar y crear para aprovisionar una máquina virtual con parámetros predeterminados o bien revise cada una de las pestañas para personalizar la configuración. [Más información](#)

Figura 30. Ingreso a los datos básicos de la máquina virtual

Se efectúa a configurar datos, en la cual en la opción de nombre podemos poner el que el usuario desee en este caso utilizamos server, el resto de opciones por default.

Crear una máquina virtual ...

carpetas para organizar y administrar todos los recursos.

Suscripción * ⓘ Azure subscription 1

Grupo de recursos * ⓘ (Nuevo) server_group
[Crear nuevo](#)

Detalles de instancia

Nombre de máquina virtual * ⓘ server ✓

Región * ⓘ (US) Centro-Sur de EE. UU.

Opciones de disponibilidad ⓘ No se requiere redundancia de la infraestructura

Imagen * ⓘ Ubuntu Server 18.04 LTS - Gen1
[Ver todas las imágenes](#)

Tamaño * ⓘ Standard_D2s_v3 - 2 vcpu, 8 GiB de memoria (80,30 US\$/mes)
[Ver todos los tamaños](#)

Figura 31. Detallando la instancia

En la cuenta del administrador que está más debajo de la ventana, se habilita la opción tipo de autenticación > contraseña y se procede a escribir un usuario y una clave, hecho esto se selecciona la opción de revisar y crear que se encuentra en la parte inferior.

Cuenta de administrador

Tipo de autenticación ⓘ Clave pública SSH
 Contraseña

Nombre de usuario * ⓘ valencia ✓

Contraseña * ⓘ ✓

Confirmar contraseña * ⓘ
✗ La contraseña y la confirmación de la contraseña deben coincidir.

Figura 32. Creando usuario y contraseña a la máquina virtual

Creado ya la máquina virtual (MV) se abrirá una nueva ventana donde podemos ir al recurso o crear otra MV, seleccionamos recursos.



Figura 33. Selección de recursos para configuración de la MV.

Ingresando a recursos se copiara la dirección IP pública y se pega en la barra de direcciones de un buscador de internet para proceder a ingresar y configurar.

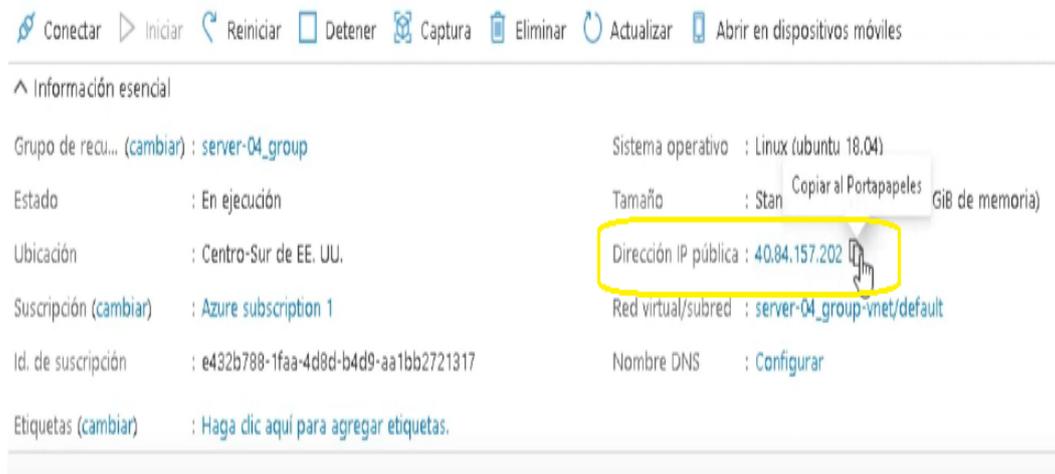


Figura 34. Obteniendo la IP pública de la MV

Luego se descarga un administrador portable de sistemas operativos para poder configurar nuestra IP, en este caso se utiliza MobaXterm en la cual ya descargado se procede a abrir y pegar la IP pública de la MV y se selecciona nueva sesión.

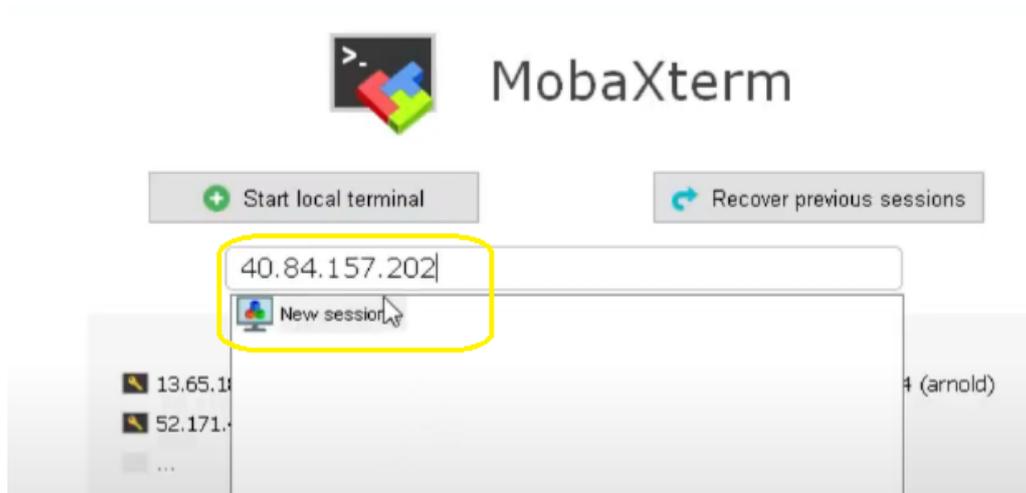


Figura 35. Configurando la IP publica en el administrador de sistema operativo

Ya ingresado se abrirá una nueva ventana la cual procedemos a ingresar a nuestro Secure Shell (SSH), la cual se procede a copiar el host remoto para utilizarlo y se selecciona la opción ok que se encuentra en la parte inferior de la ventana.

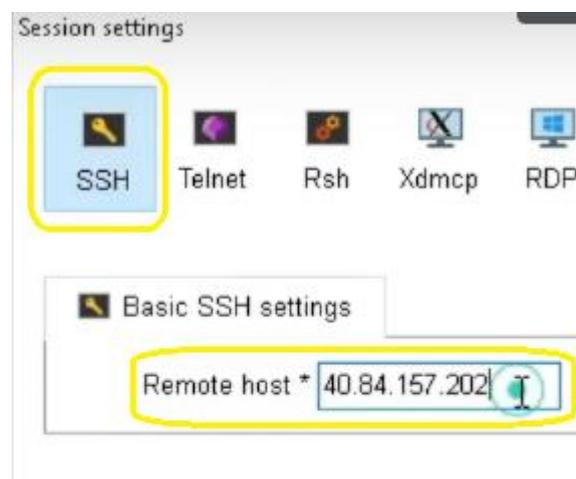


Figura 36. Seleccionando el host remoto del SSH.

Hecho esto aparecerá un CMD la cual pedirá nuestro usuario y clave que se ha creado como se muestra en la Fig. 32.

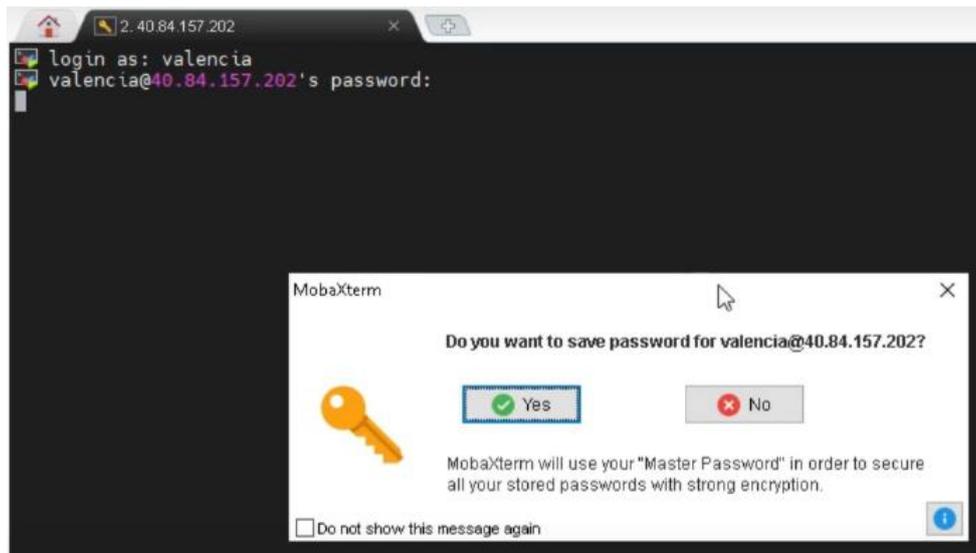


Figura 37. Ingresando a la máquina virtual

Luego se procederá a ingresar el comando sudo su para rutiario el servidor.

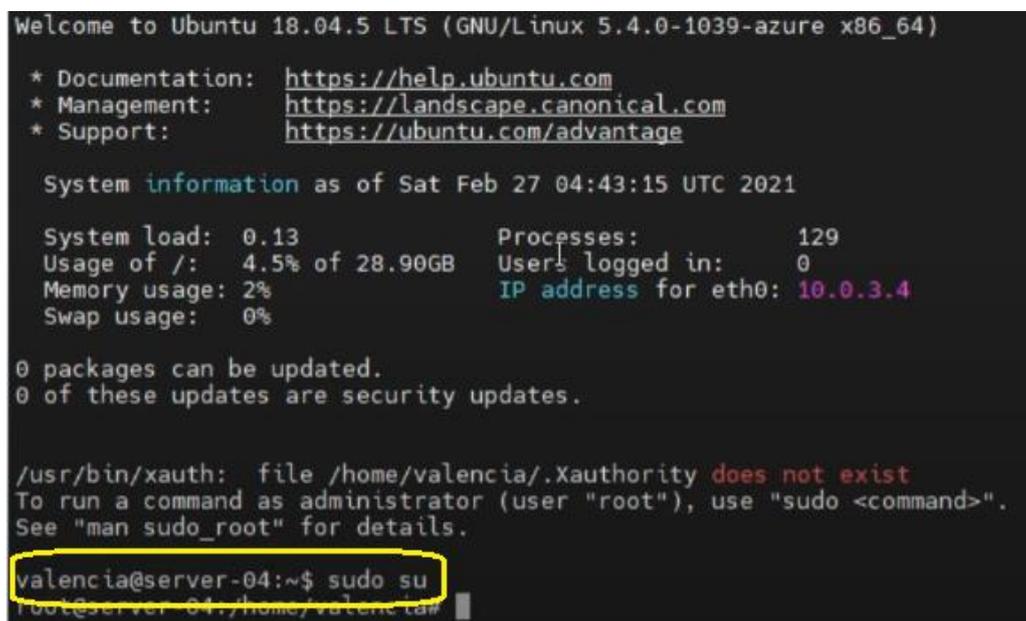


Figura 38. Rutiario el servidor de la MV

Rutiario el servidor se procederá a ingresar algunos comandos para activar la MV, la cual se escribe y se da enter.

Comandos para activar la MV:

apt-get update [40]

apt-get install ubuntu-desktop [40]

apt-get install --reinstall ubuntu-desktop [40]

nano script1.sh [40]

chmod +x script1.sh [40]

./script1.sh [40]

apt install xfce4 slim [40]

service slim start [40]

```
aspell-autobuildhash: processing: en [en_CA-wo_accents-only].
aspell-autobuildhash: processing: en [en_GB-ise-w_accents-only].
aspell-autobuildhash: processing: en [en_GB-ize-w_accents-only].
aspell-autobuildhash: processing: en [en_GB-ize-wo_accents-only].
aspell-autobuildhash: processing: en [en_GB-variant_0].
aspell-autobuildhash: processing: en [en_GB-variant_1].
aspell-autobuildhash: processing: en [en_US-w_accents-only].
aspell-autobuildhash: processing: en [en_US-wo_accents-only].
Processing triggers for initramfs-tools (0.130ubuntu3.11) ...
update-initramfs: Generating /boot/initrd.img-5.4.0-1039-azure
Processing triggers for libgdk-pixbuf2.0-0:amd64 (2.36.11-2) ...
Processing triggers for libc-bin (2.27-3ubuntu1.4) ...
Processing triggers for dbus (1.12.2-1ubuntu1.2) ...
Processing triggers for ureadahead (0.100.0-21) ...
Processing triggers for systemd (237-3ubuntu10.44) ...
root@server-04:/home/valencia# apt-get install --reinstall ubuntu-desktop
```

Figura 39. Instalando comandos para activación de la MV

Ya instalados todos los comandos escritos se escribe el comando reboot y se cierra el tab.

```

Processing triggers for initramfs-tools (0.130ubuntu3.11) ...
update-initramfs: Generating /boot/initrd.img-5.4.0-1039-azure
root@server-04:/home/valencia# service slim start
root@server-04:/home/valencia# reboot

Remote side unexpectedly closed network connection

Session stopped
- Press <return> to exit tab
- Press R to restart session
- Press S to save terminal output to file

```

Figura 40. Reboteando la MV y finalizando el proceso activación

Concluido esto si se hizo bien el proceso de escribir los comandos, procedemos a dirigimos a Azure>diagnóstico de arranque y nos aparecerá una imagen del proceso de comandos para activar el MV ya efectuado.

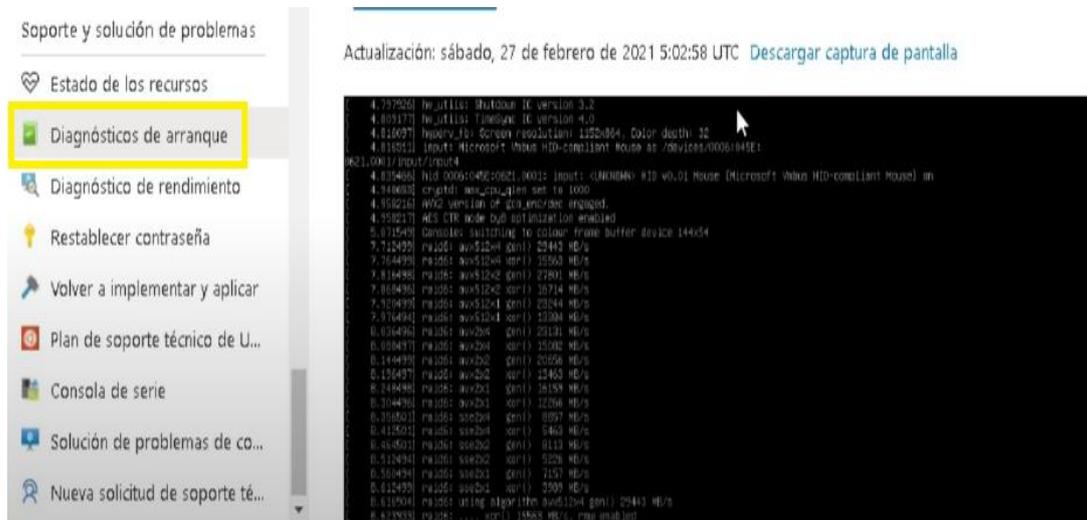


Figura 41. Comprobando el proceso de activación

Comprobando luego nos dirigimos a configuración y vamos a redes> agregar regla de puerto de entrada.

Hecho esto se abrirá una ventana para configurar algunos parámetros como: en la opción servicio seleccionamos RDP, en acción permitir y en el nombre los mismo RDP; procedemos a finalizar seleccionando agregar

Acción

Permitir

Denegar

Prioridad * ⓘ

310

Nombre *

RDP ✓

Descripción

Figura 42. Configurando la regla de seguridad de entrada

Se abre de nuevo Mobaxterm para visualizar nuestra máquina virtual y procedemos a hacer el mismo procedimiento como en la Figura 36. ingresamos la IP y se procede a dirigir a RDP y escribimos nuestro usuario, contraseña y seleccionamos ok.

Basic Rdp settings

Remote host *

Username

Figura 43. Ingresando y visualizar la MV

Hecho esto ya se podrá ingresar a nuestra máquina virtual en Azure y se procederá a configurar lo que se desea como si fuera un ordenador.

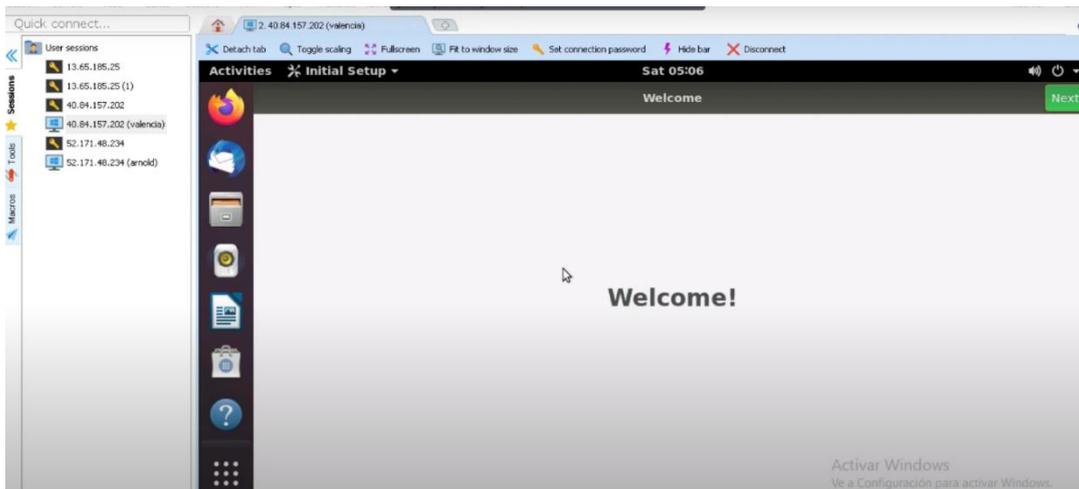


Figura 44. Máquina virtual en Azure

3.11. NODE RED

Se accede al URL y ahí se procede a poner el usuario y clave para iniciar el nodo red, se procede a seleccionar siguiente y comenzara a instalarse.

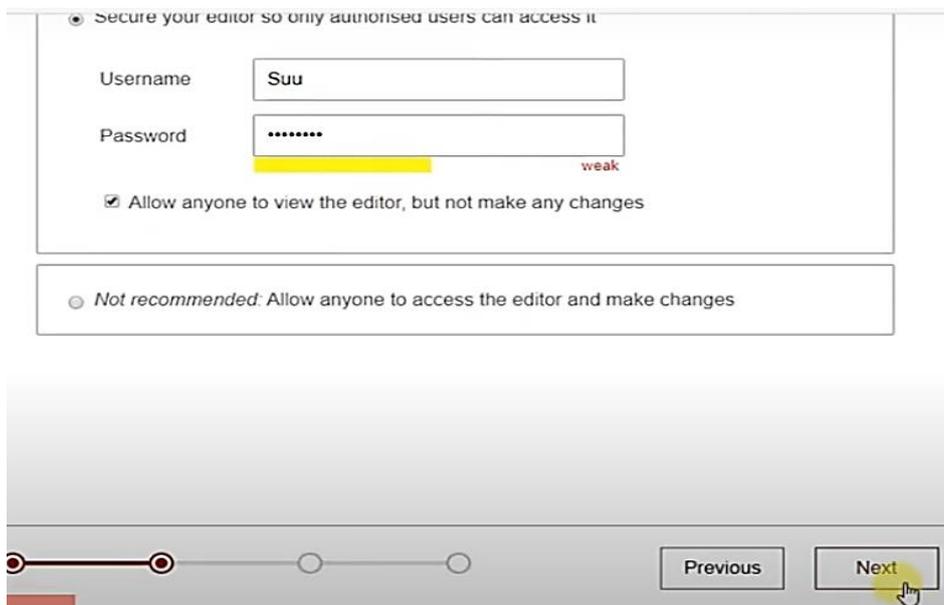


Figura 45. Generando usuario y contraseña del Node red.

Ya instalado se puede iniciar Node red y se escogerá la opción ir a editor de Node red (go to your node red flow editor).

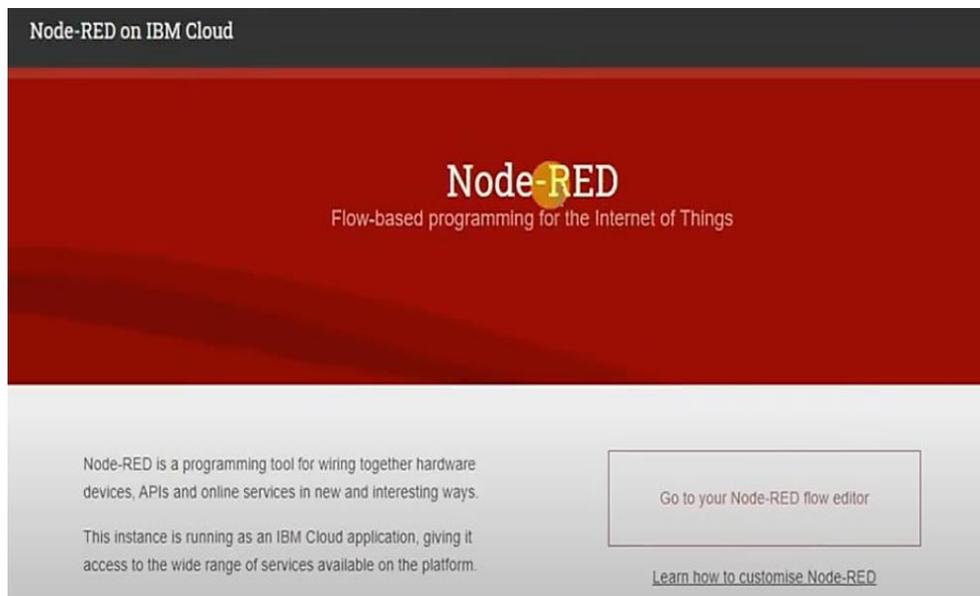


Figura 46. Ingresando a node red

Ingresando va a pedir usuario y contraseña como se ve en la Figura 24 se ingresa a nuestro editor de Node red.

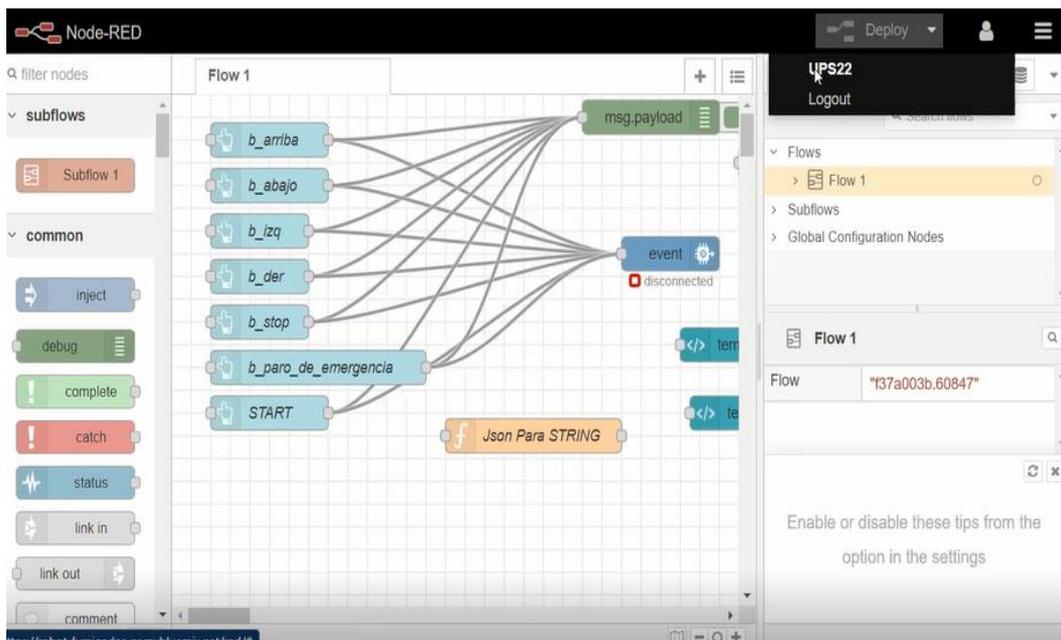


Figura 47. Ingresando al editor de Node red

Para proceder a hacer el flujo de programación en el Node red, como primer instancia se va al menú de node red que está en la parte superior derecha de la ventana y seleccionamos en administrador de paleta (Manage palette), la cual se podrá instalar IBM Watson e IBM IOT, que son flujos o nodos importantes para poder comunicarse al Node red local, que está instalado en el mini ordenador Raspberry y que a su vez se comunica a la nube que habilita por medio de IBM.

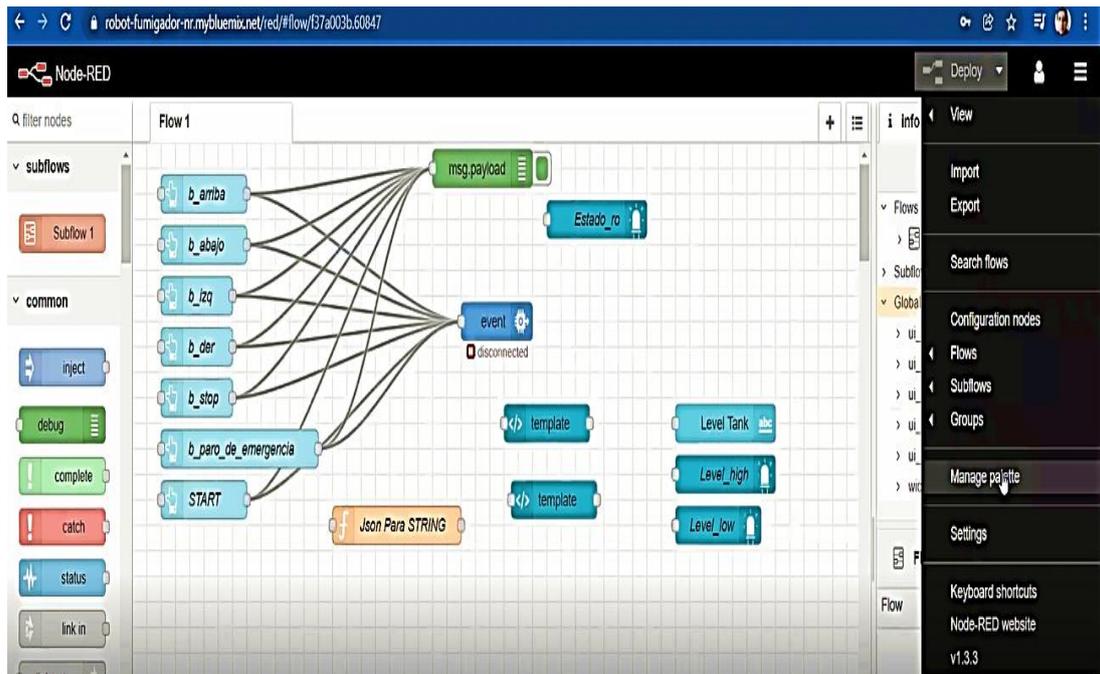


Figura 48. Ingresando al administrador de paleta de Node red.

El Manage Palette es la herramienta, que sirve para ver que nodos están instalados en el Node red y también sirve para instalar los nodos que se requiriera de acuerdo a las necesidades que el usuario vaya a resolver, en este caso se procede a buscar el flujo Watson en nuestro buscador e instalamos el nodo: node-red-contrib-ibm-watson-lot la cual se hace dando click en instalar.

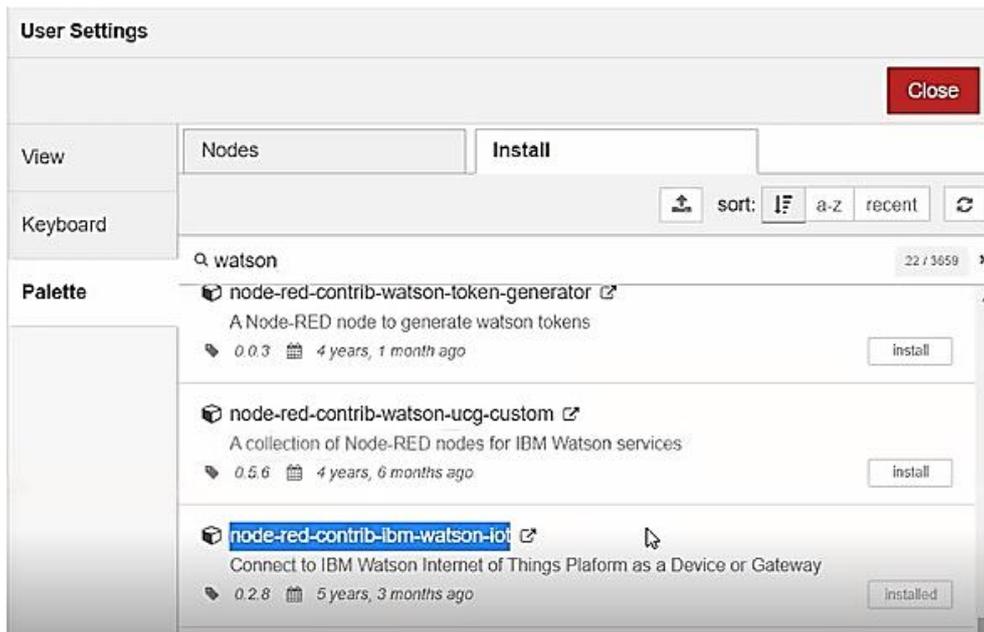


Figura 49. Instalando IBM Watson al Node red.

En el nombre del nodo o flujo que se desea instalar si se selecciona el icono cerca del nombre nos enviara a un link en la cual se puede ver sus características como: instalación en el Raspberry, como usar el nodo, características técnicas, entradas y salidas,

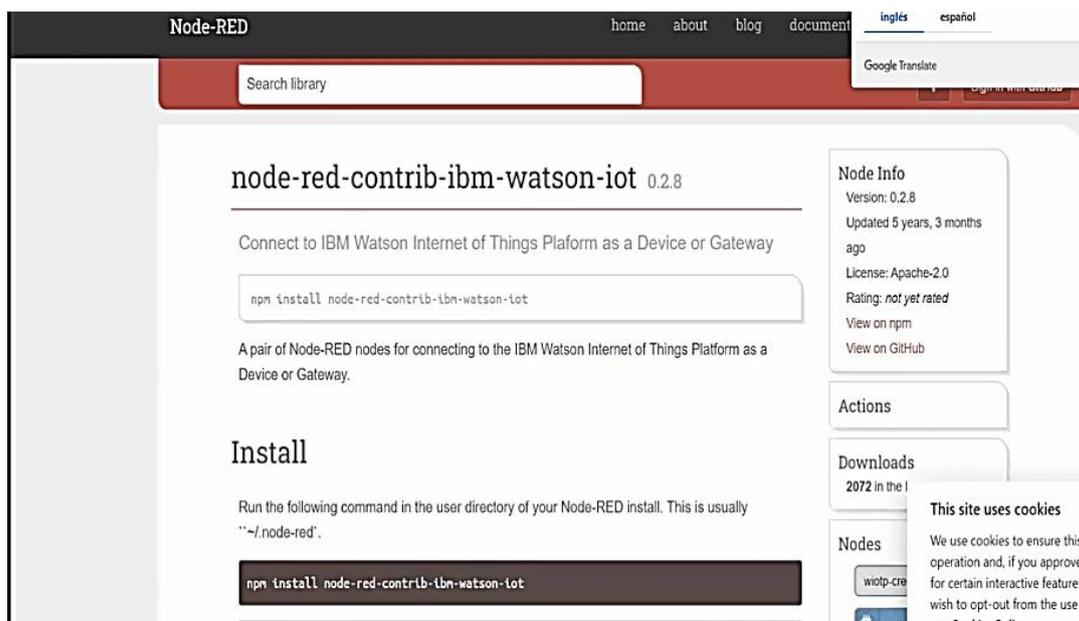


Figura 50. Características del nodo instalado

A continuación, se procede a hacer los mismos pasos como se explica en la Figura 37. Pero ahora buscando el nodo IBM IOT, lo buscamos e instalamos el nodo (node-red-contrib-scx-ibmiotapp), para saber su ficha técnica se da click en el icono a lado del nombre del nodo como se explicó en la figura anterior. Una vez instalados nuestros nodos damos cerrar a nuestro Manage Palette

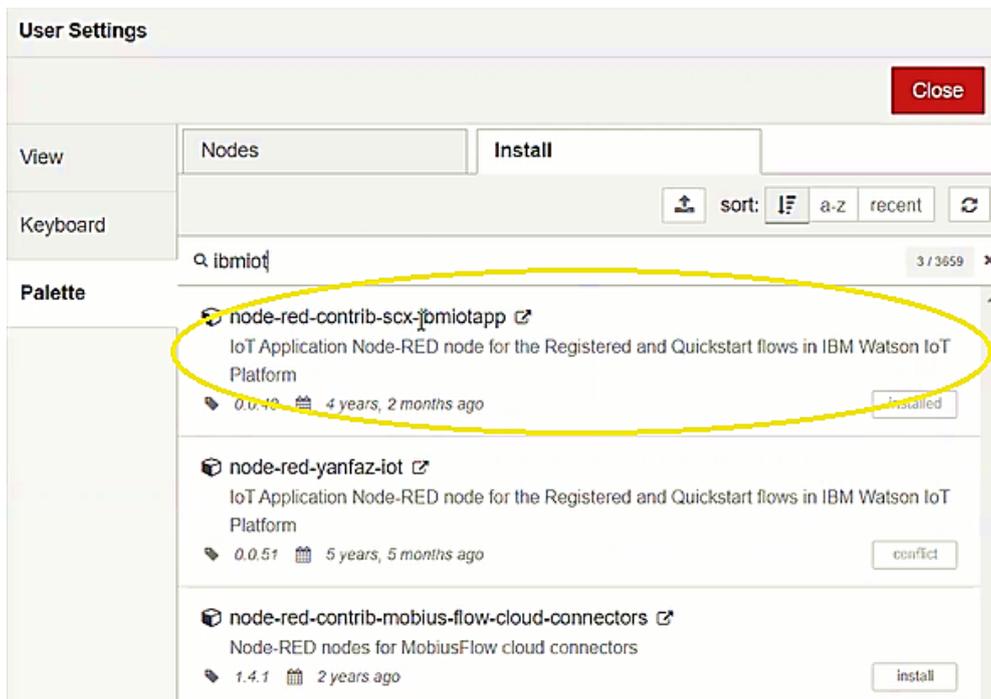


Figura 51. Instalando Node IBM IOT a nuestro Node red

Para verificar si están instalados correctamente nuestros nodos, nos dirigimos a nuestros filtros de nodos que está ubicado en la sección izquierda de nuestra ventana de Node red y bajamos hasta entradas y salidas (input y output).

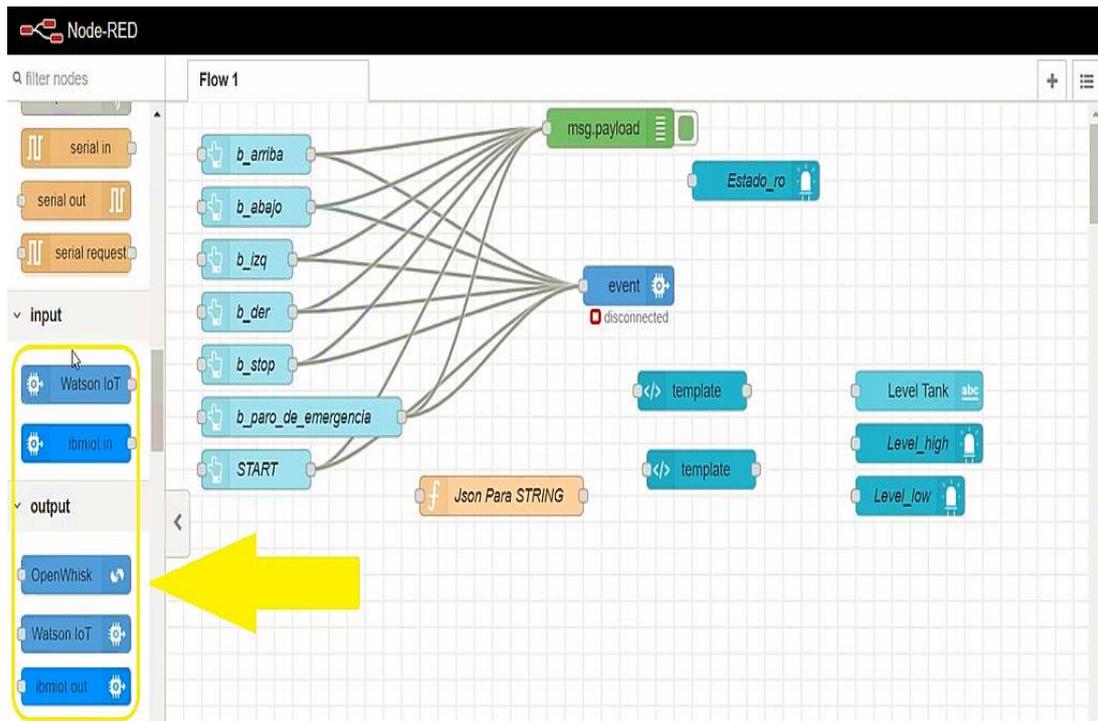


Figura 52. Verificación de instalación de nodos

Esta instalación de los nodos, se debe efectuar tanto en el Node red que está en la plataforma IBM (nube) y en el Raspberry (local). por medio de VNC Viewer para que exista una comunicación bidireccional.

Una vez hecho esto, volvemos a la página de IBM y nos dirigimos al panel de control IBM como ya se mostró en la Figura 29. Y accedemos a la lista de recursos.

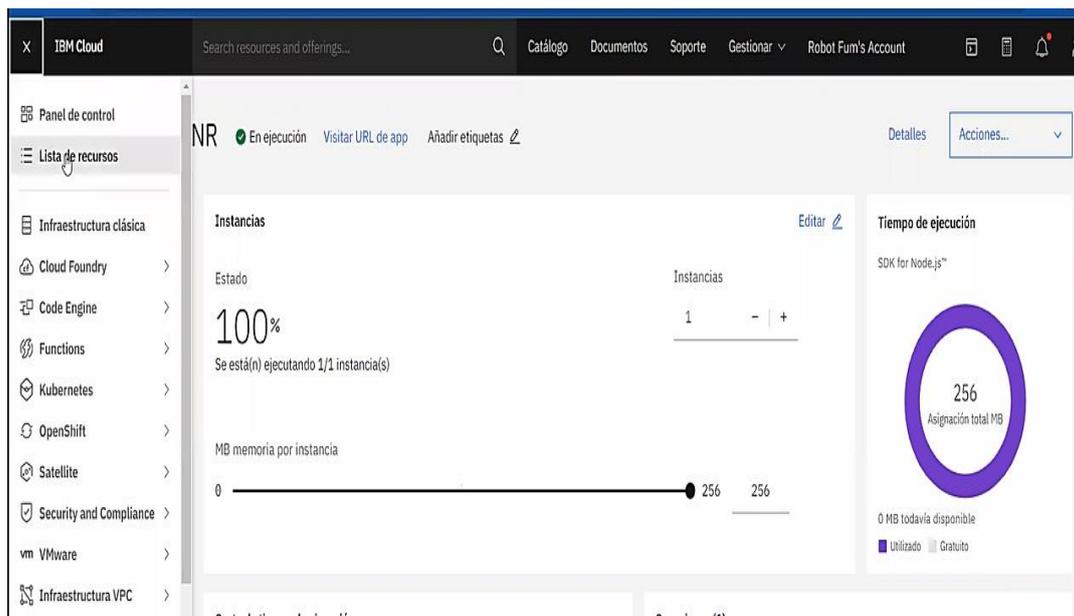
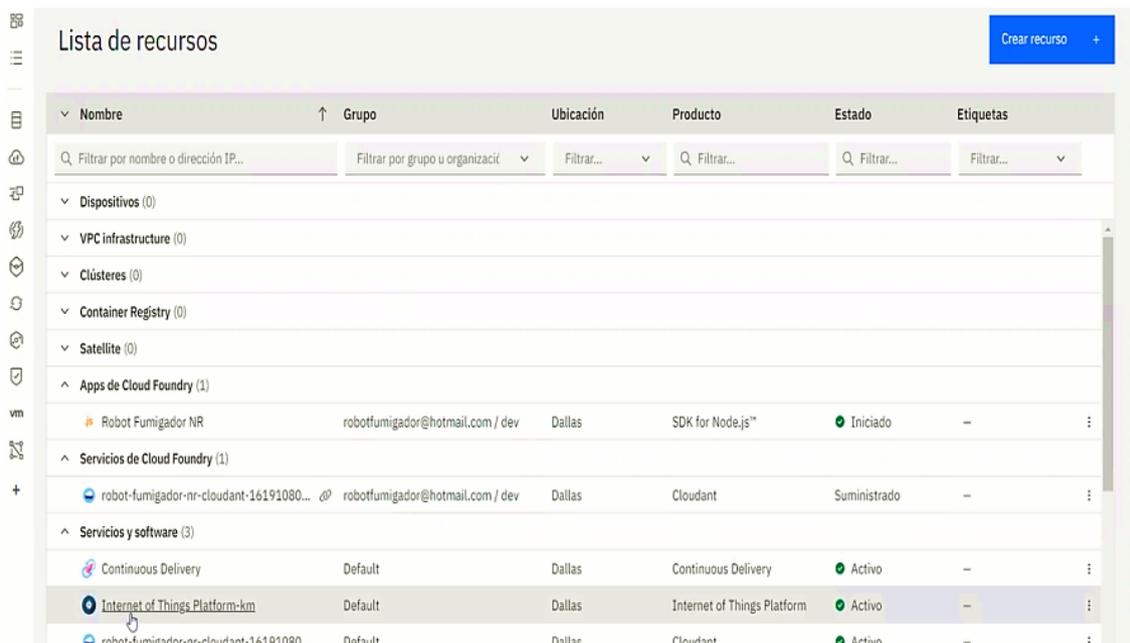


Figura 53. Entrando a la lista de recursos de IBM Cloud.

Seguimos al servicios y software, seleccionamos internet of thing ya instalado con anterioridad como se muestra en la Figura 20. Si no está ese servicio IoT en nuestra pestaña, significa que ha expirado ya que estos servicios se inhabilitan si se da inactividad por 30 días ya que son gratuitos y se tiene que volver a instalar este servicio a nuestro IBM.



The screenshot shows the 'Lista de recursos' (Resource List) interface in IBM Cloud. It features a table with columns for Nombre, Grupo, Ubicación, Producto, Estado, and Etiquetas. The table is filtered to show resources under the 'Servicios y software' category. The 'Internet of Things Platform-km' resource is highlighted, indicating it is active.

Nombre	Grupo	Ubicación	Producto	Estado	Etiquetas
Dispositivos (0)					
VPC infrastructure (0)					
Clústeres (0)					
Container Registry (0)					
Satellite (0)					
Apps de Cloud Foundry (1)					
Robot Fumigador NR	robotfumigador@hotmail.com / dev	Dallas	SDK for Node.js™	Iniciado	—
Servicios de Cloud Foundry (1)					
robot-fumigador-nr-cloudant-16191080...	robotfumigador@hotmail.com / dev	Dallas	Cloudant	Suministrado	—
Servicios y software (3)					
Continuous Delivery	Default	Dallas	Continuous Delivery	Activo	—
Internet of Things Platform-km	Default	Dallas	Internet of Things Platform	Activo	—
robot-fumigador-nr-cloudant-16191080	Default	Dallas	Cloudant	Activo	—

Figura 54. Ingreso al servicio IoT IBM.

Ingresado nos llevara a nuestra ventana de IBM Watson la cual seleccionamos la opción lanzar.



Figura 55. Ejecutando IBM Watson IOT platform.

Ya ejecutado nos llevará a otra ventana, que será la plataforma de IBM Watson la cual vamos a escoger nuestra organización que se encuentra superior derecha y procedemos a escoglarla.

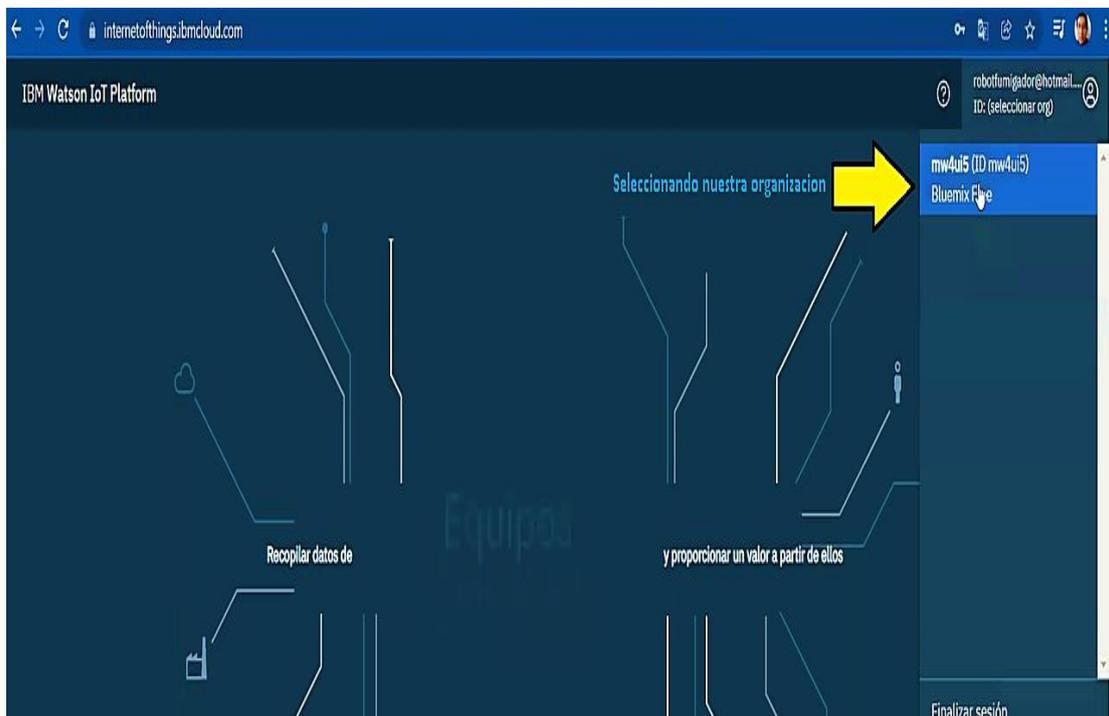


Figura 56. Plataforma IBM Watson y selección de la organización.

Seleccionado nos va a llevar a IBM Watson, la cual se podrá observar todos los dispositivos creados, como ya tenemos creado uno que se puede apreciar en nuestra Figura 26.

Volvemos al Node red y arrastramos nuestro nodo de salida de Watson IOT al scrip de programación.

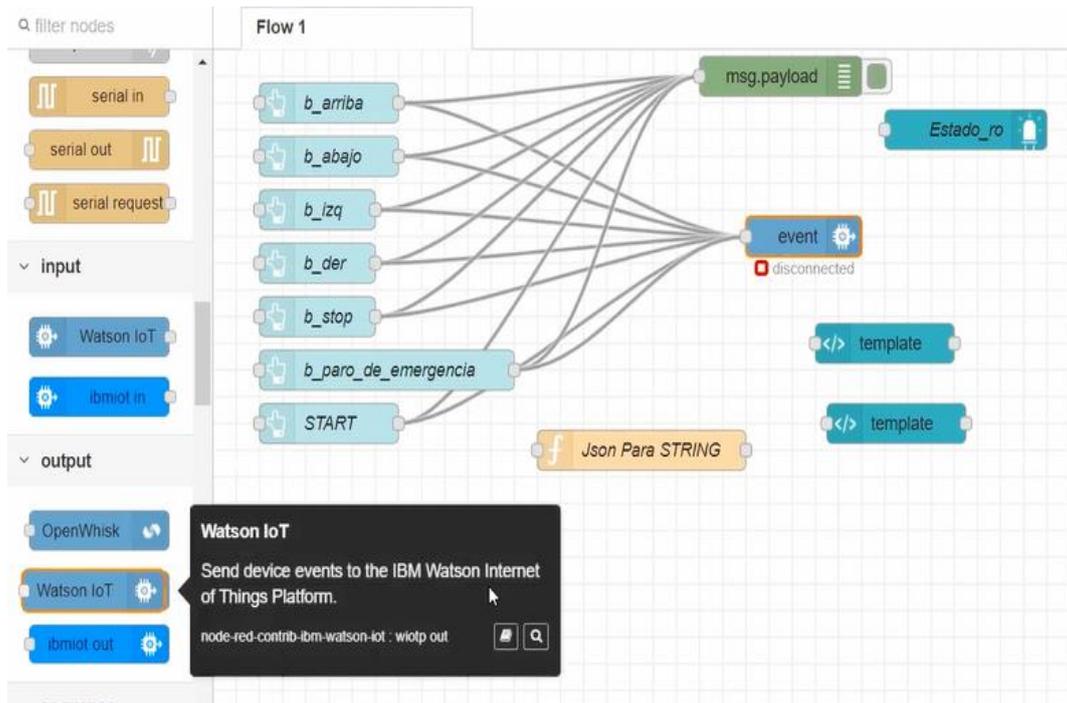


Figura 57. Seleccionando Node Watson IOT de salida al scrip de Node red.

Realizado esto, seleccionamos el nodo para entrar al menú de edición en la cual se abrirá una nueva ventana donde dará algunas opciones para configurar, en la opción conectar a (connect as) escogemos la opción dispositivo (device), en la opción registro (register) damos click para que aparezcan las credenciales ya creadas como se muestra en la figura 27. En el tipo de eventos y formato; se selecciona configuración de credencial que es un icono que se encuentra superior derecha de este.

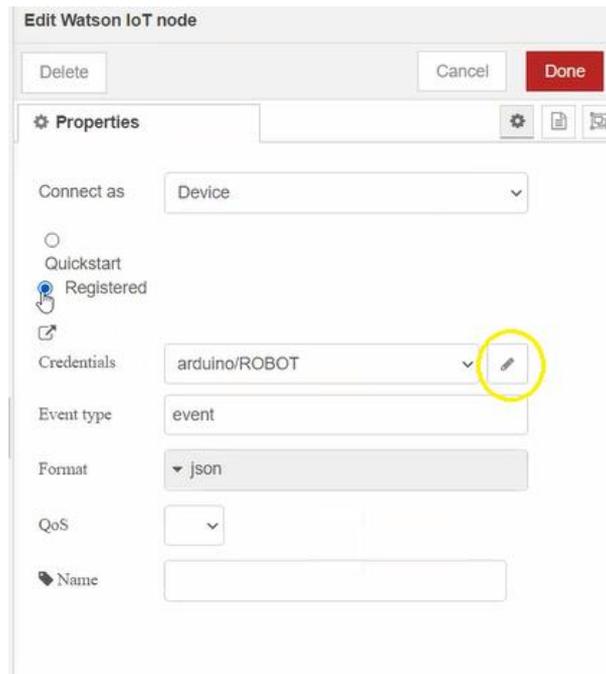


Figura 58. Editando Node Watson IoT.

Para llenar el bloque de organización se procede a dirigir a la página IBM específicamente a la plataforma IBM Watson, copiamos la organización creada como se puede apreciar en la Figura 44. Y se procede a pegar en el bloque que corresponde al nodo Watson en Node red, se procede a hacer el mismo procedimiento para las opciones de nombre del dispositivo e ID del dispositivo que podemos encontrar esa información en la Figura 21. Procedemos a pegar y se obtiene como el tipo de dispositivo: Arduino, y en ID dispositivo: ID.

En el caso de autenticación del token se procede a escribir la clave que se creó en el dispositivo de IBM Watson como se observa en la Figura 28. Una vez hecho esto se selecciona la opción agregar (update) y hecho (done).

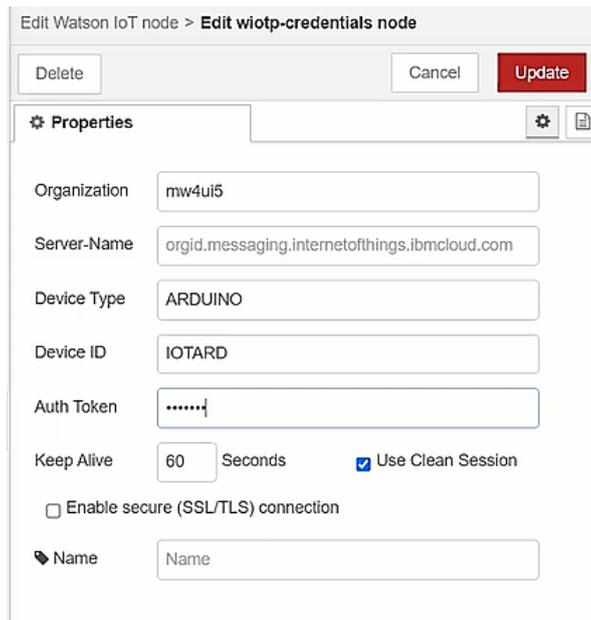


Figura 59. Editando credenciales del nodo Watson IoT.

Se selecciona nuestro Nodo Watson IoT y se procede a hacer una prueba respectiva si hay conectividad, nos dirigimos a la paleta de filtros de nodos que está en la parte izquierda de la ventana. Se procede a dirigir hasta la sección de dashboard (tablero de mandos) y seleccionamos el nodo button a nuestro scrip.

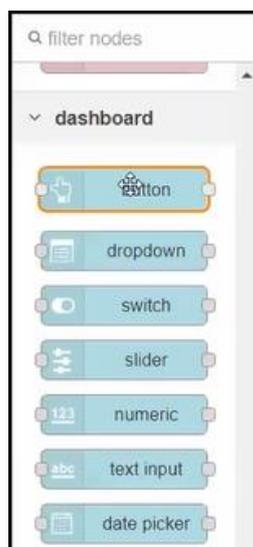


Figura 60. Selección de nodo button para el dashboard.

Se procede a dar doble click al botón para entrar a la edición del botón, en la cual se dará varias opciones a editar como: grupos, tamaño (size), iconos, etiqueta (label), tips de herramientas (tooltips), color, etc. En la opción grupos se puede escoger a que grupo de tablero se quiere pertenecer en este caso el grupo 1 y en tamaño por default.

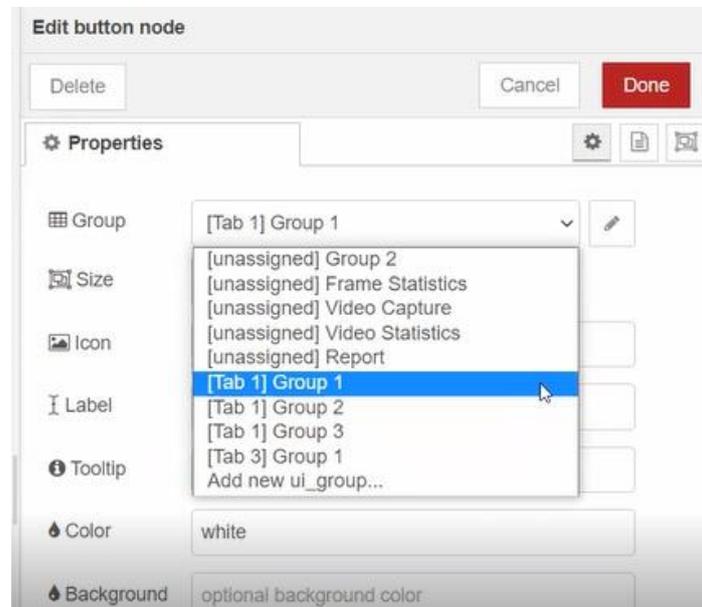


Figura 61. Editando nodo button la propiedad del grupo.

En la opción icono se desea tener una perspectiva más agradable para el dashboard o tablero; como se desea que sea fácil de entender al usuario y con buena presentación se procederá a descargar una imagen que indique la acción que se desea realizar, para esto se busca una librería en la web, en este caso se descargara de la página fa-chevron-up la cual existen diferentes tipos de iconos para utilizar en el dashbord.

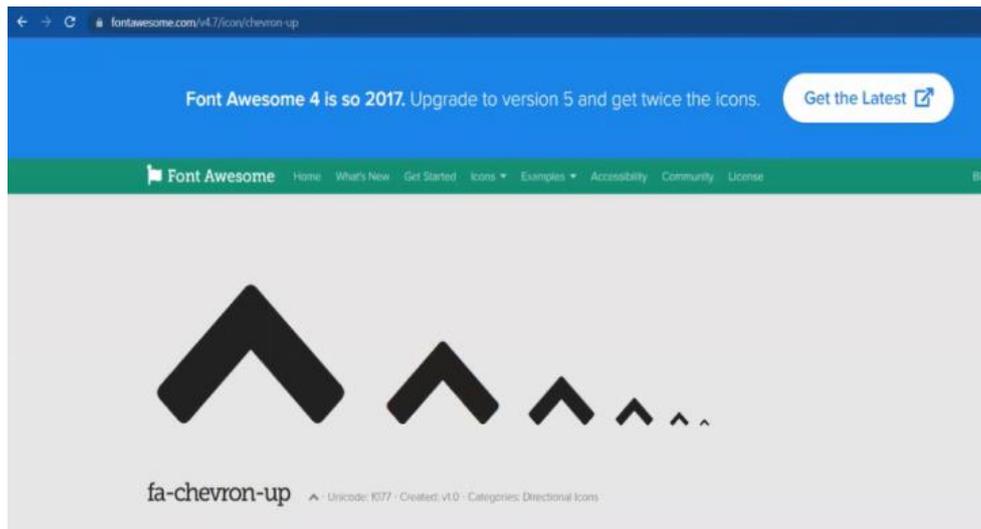


Figura 62. Descargando librería de iconos para los nodos button del dashboard

Se adjunta el archivo descargado en la opción icono de edición del botón (fa-chevron-up fa-3x) la cual 3x indica el tamaño del icono; el color se escogió el blanco y en tooltip se escribe it forward (adelante) que esta opción es una herramienta de ayuda que sirve para indicar que hace cada elemento del dashboard,

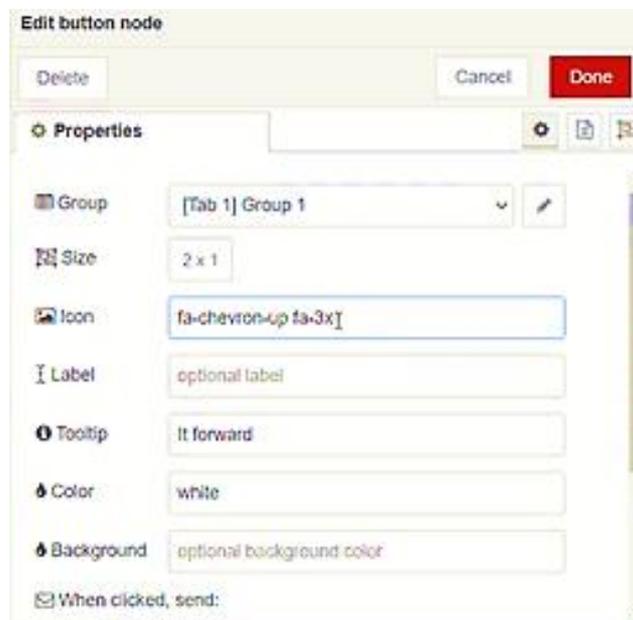


Figura 63. Editando las demás opciones del node button del dashboard

Se sigue con la edición del botón y se selecciona la opción cuando des click (when

clicked), en la cual ahí se seleccionará que tipo de variable puede ser para el botón, como, por ejemplo: string, booleano, flotante, etc. En este caso se va a utilizar tipo string ya que esto servirá más adelante con la comunicación de la programación con el microprocesador Arduino.

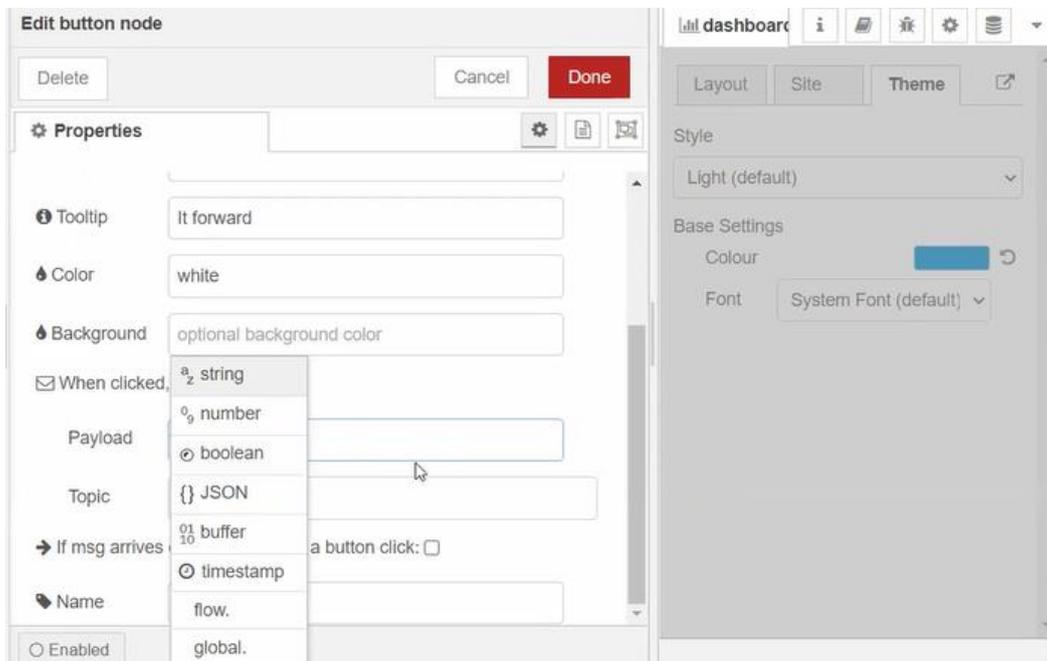


Figura 64. Editando nodo button el tipo de variable.

Escogiendo la variable, se procede a dar un valor al botón en la opción payload, en este caso como es tipo letra (string) se da como valor la letra "a", significa que cada vez que se pulse el botón se va a enviar un string con valor "a". En la opción name se escoge el nombre que se quiere poner al nodo en este caso b_arriba (botón arriba) y en las demás opciones del editor de botón como default.

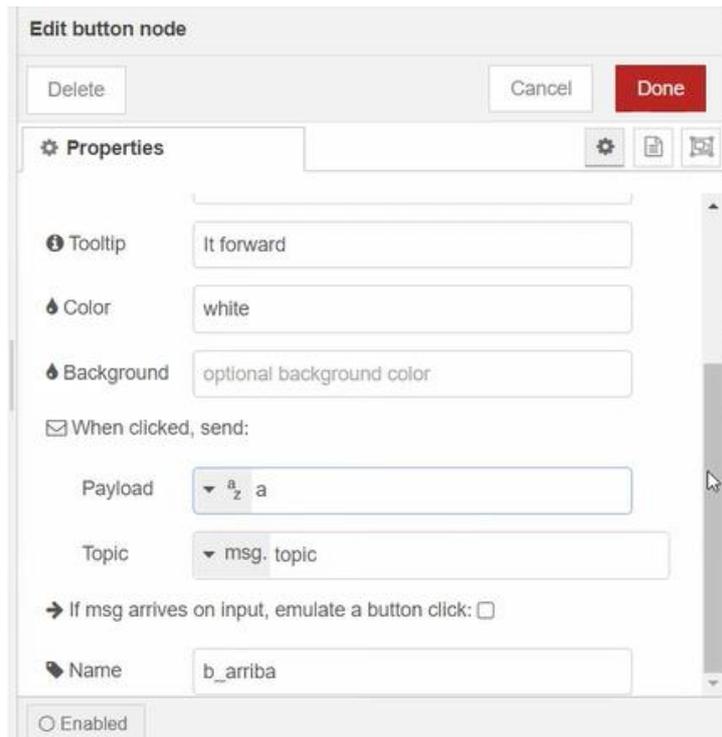


Figura 65. Editando el nodo button (dando valor alfabético y nombre a la variable).

Se procede a escoger 3 botones más para el mando de abajo, derecho e izquierda y se procede a hacer las mismas instrucciones como se explica en la Figuran 41 hasta la Figura 53.

Realizado esto se procede a poner un bloque de nodo llamado msg.payload; es un nodo debug (depurador), la cual sirve para poder visualizar que carácter se está enviando en los botones por medio del Node red, este nodo se encuentra en el filtro de nodos que está en la ventana izquierda

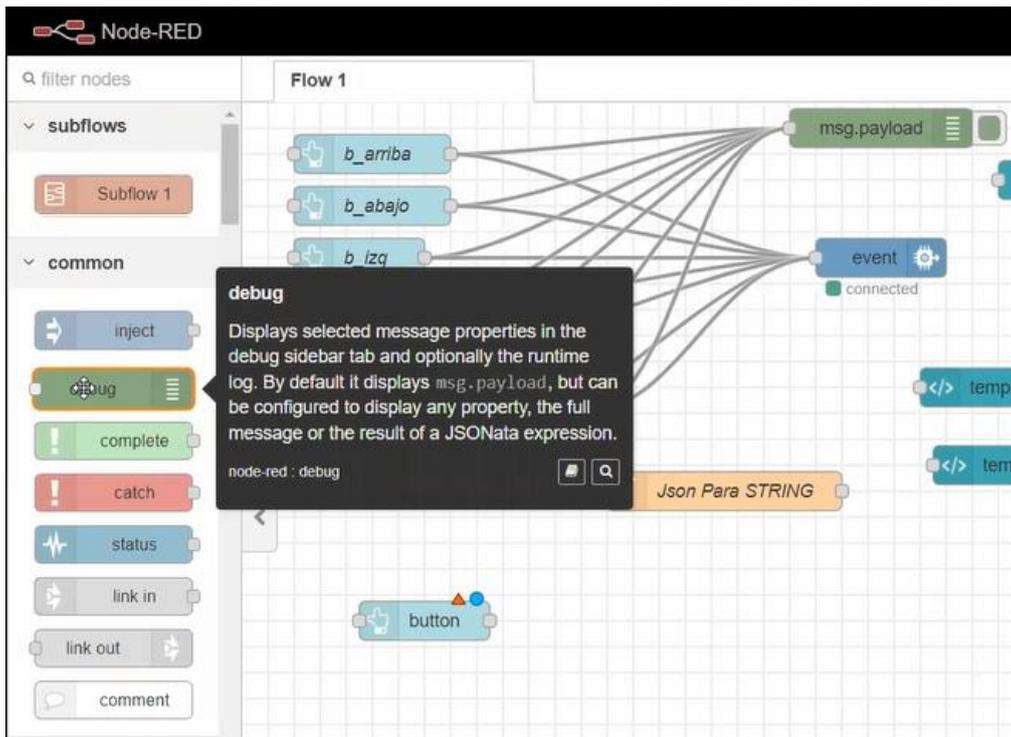


Figura 66. Seleccionando el nodo msg.payload.

Seleccionado y configurado los botones, se procede a conectar el nodo Watson IOT para hacer las respectivas pruebas, en este caso se lo tiene con el nombre de evento.

A continuación, damos click deploy que se encuentra superior izquierda de nuestro Node red para correr el programa.

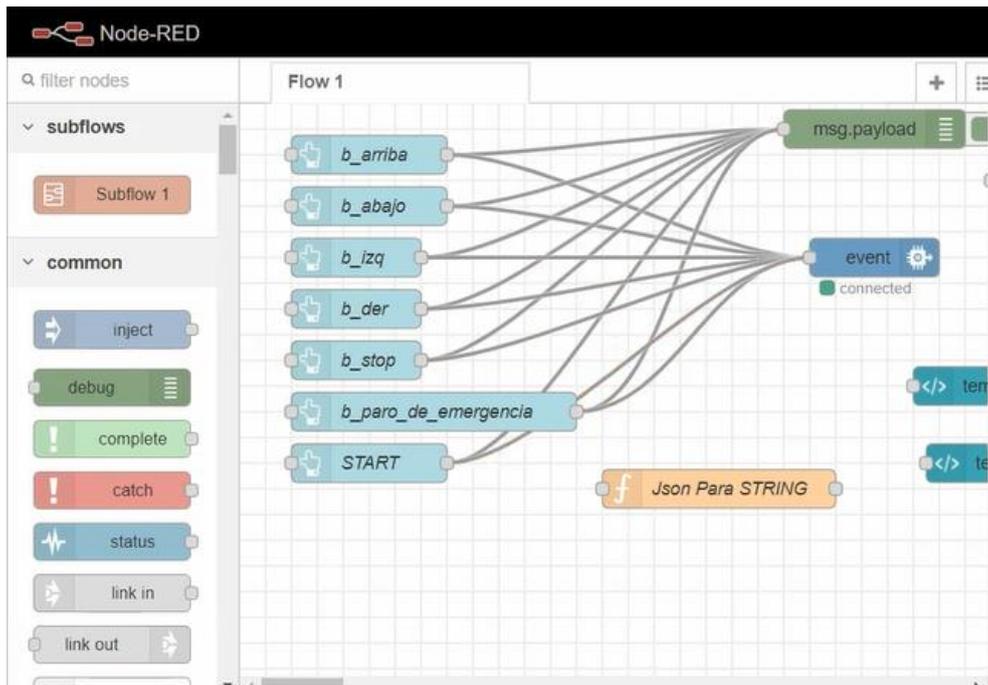


Figura 67. Conexión de los nodos buttons al nodo Watson IoT

Se procede a escoger otro nodo de la paleta de filtros que servirá para la comunicación de la cámara, la cual es el nodo template, se selecciona y se lo lleva al scrip.

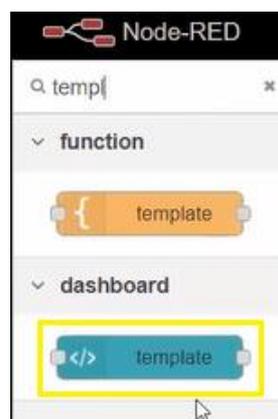


Figura 68. Selección de nodo template para comunicación de cámara IP.

Entramos al nodo template dando doble click, la cual sirve para escoger un archivo tipo html y se podrá configurar el tamaño de pantalla como es ancho y alto; además del src que es un link la cual se encarga de leer la proyección que está enviando el

dispositivo de audiovisual.

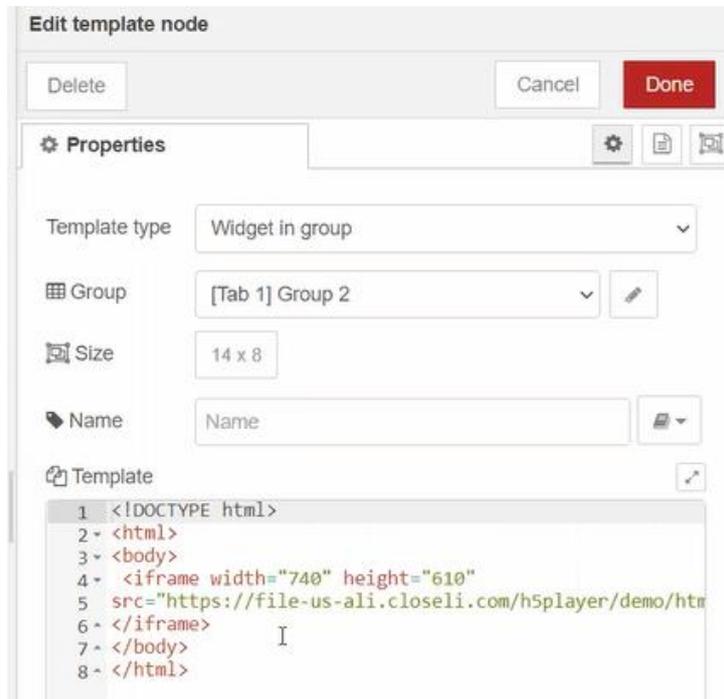


Figura 69. Configuración del nodo template

Si siguiendo bien los pasos indicados saldrá un mensaje de conectado debajo del nodo Watson IoT. Para poder ingresar al dashboard, se selecciona la opción de herramientas que está superior derecha de la ventana del Node red que es el símbolo de una flecha la cual brindará varias opciones como información, ayudas, mensajes de depuración, configuración de nodos, contexto de datos y el dashboard. Una vez ingresado se podrá observar nuestros botones instalados.

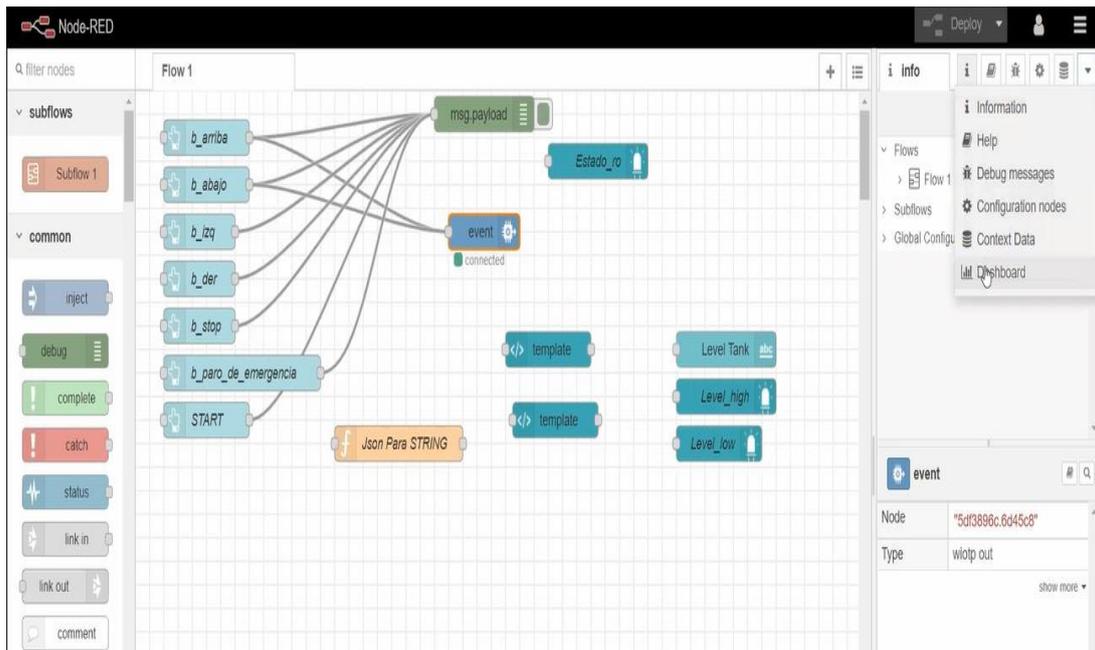


Figura 70. Ingresando al dashboard.

Va a salir en la ventana en la parte derecha el diseño, sitio y temas de dashboard. En diseño podemos ver los grupos de tabs y links, la cual sirven para separar por grupos de ventanas los botones; en este caso vamos a modificarlo.

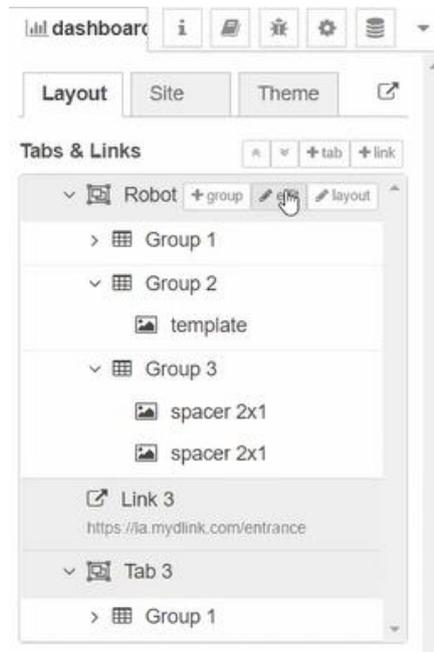


Figura 71. Editando el dashboard.

Hecho esto, se abrirá una ventana en la cual se podrá modificar el nombre al grupo, esto sirve para que la ventana del dashboard cuando ejecutamos de Node red salgo el nombre que se escogió.

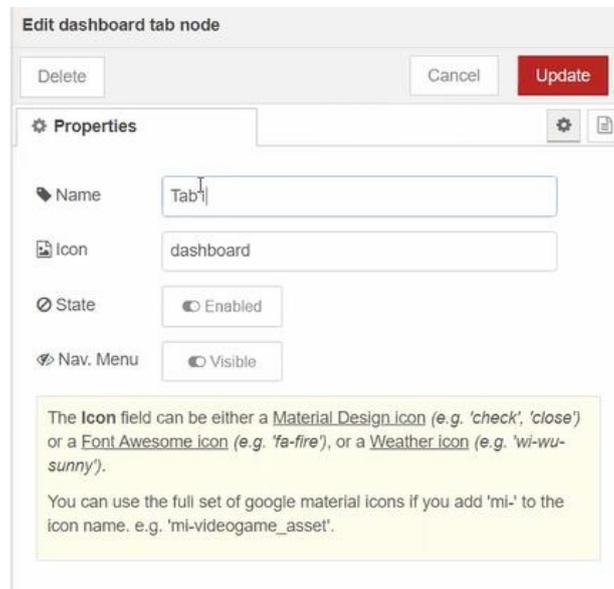


Figura 72. Configurando el tab del dashboard

Dentro de cada tab, se obtiene los grupos que son una serie de botones o actuadores, que se utilizan para una correcta organización visualizada del dashboard, en este caso se creó dos grupos, uno para dividir los botones y otro para la cámara, la cual crearemos pulsando el botón de crear a lado del tab ya creado anteriormente

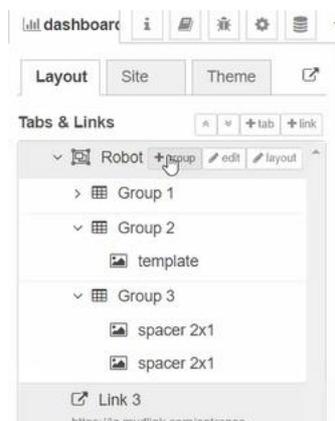


Figura 73. Creación de grupos en el tab del dashboard.

Luego se procede a escoger todos los botones del dashboard en el grupo 1 y para la conectividad del video a través de la cámara se escogió grupo 2. Para poder modificar la estructura y diseño del dashboard, se seleccionará a lado del tab en la opción layout.

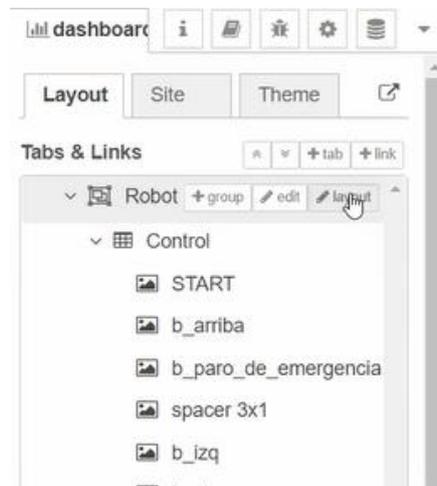


Figura 74. Ingreso de buttons en grupos e ingreso de diseño al dashboard.

Ingresado al layout, se podrá modificar los botones del dashboard ubicándolos como desea el diseñador, con solo moverlos con el mouse en la cual también mostrara cuantos grupos se tienen creados.

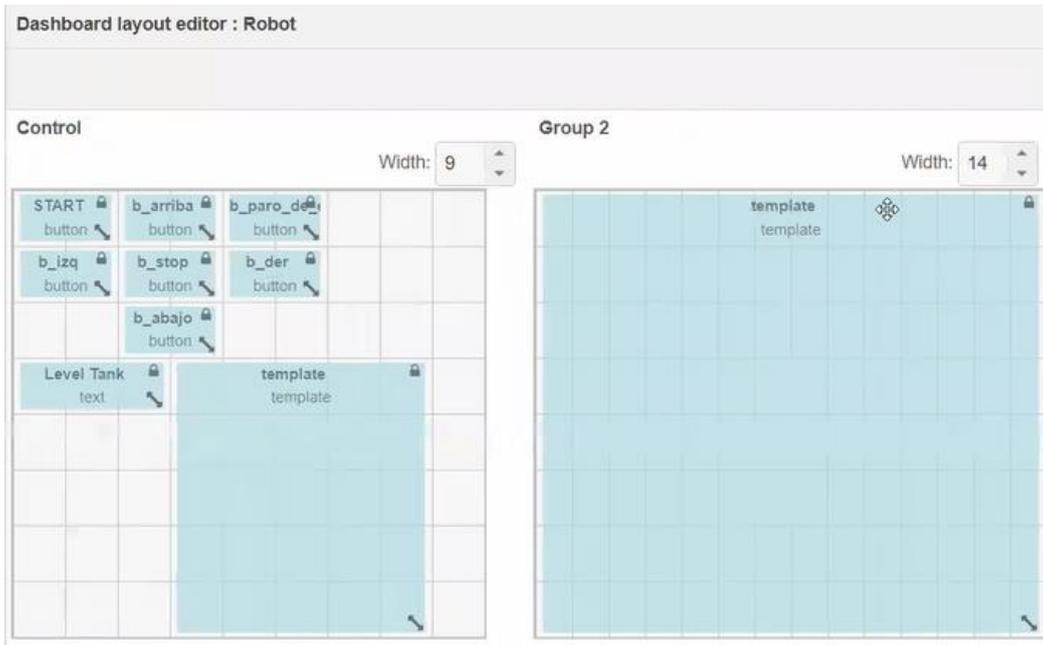


Figura 75. Diseño de la estructura del dashboard.

Hecho esto, se selecciona deploy al Node red y se abrirá la ventana del dashboard ya modificado. Si se pulsa uno de los botones en el dashboard, se visualizará en el Node red que carácter envía cada botón del dashboard por medio del nodo ya conectado que es msg.payload.

Otra verificación para saber si se enviaron correctamente los datos string es dirigirnos a la plataforma de IBM Watson IOT, se selecciona el dispositivo ya creado en IBM y se dirige a sucesos reciente, ahí se podrá apreciar los caracteres enviados del node red.

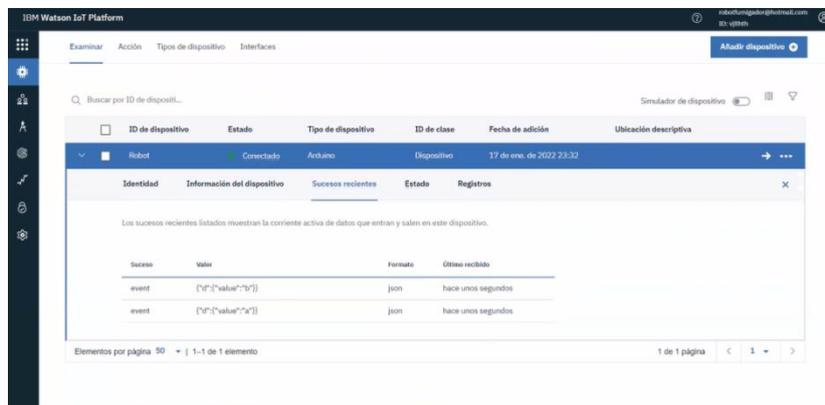


Figura 76. Verificación de caracteres enviados de node red a IBM Watson IoT.

Se dirige nuevamente a node red y se abrirá un nuevo scrip para introducir el nodo IBM IoT local, la cual se encontrara en la paleta de filtro de nodos que se encuentra en la parte izquierda de la ventana explicado con anterioridad, se avanza a la sección entradas y se selecciona al scrip tal cual como se hizo con el nodo IBM Watson como se muestra en la figura 37.

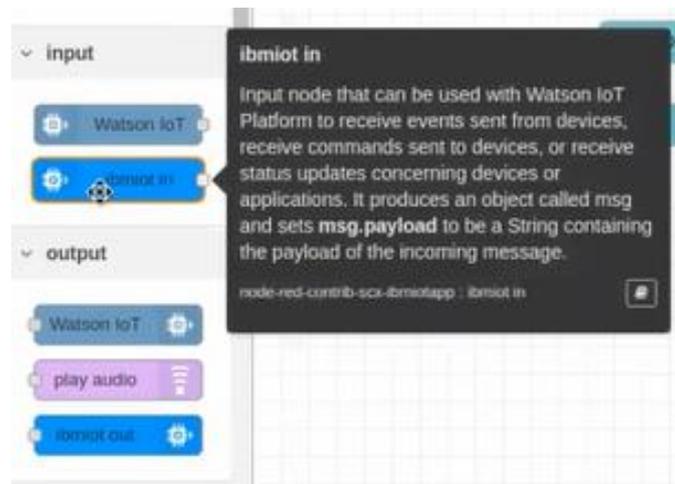


Figura 77. Selección de nodo IBM IoT

Luego se procede a dar doble click en el nodo y se abrirá una nueva ventana para la configuración, la cual habrá opciones para: la autenticación, tipo de entrada, identificación del dispositivo, nombre y servicio. En la opción autenticación se selecciona API Key porque se está utilizando una API la cual ya se ha creado como se puede apreciar en la figura 23.

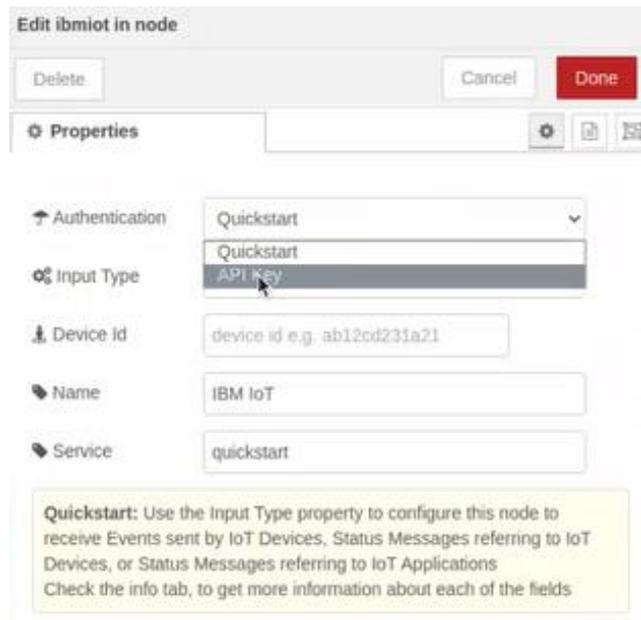


Figura 78. Configuración del nodo IBM IoT seleccionando el tipo de autenticación.

Seleccionando la autenticación, aparecerán más opciones en la configuración del nodo como clave de API, tipo de dispositivo, evento y formato.

Como primera instancia se procede a modificar la clave API, se procede a dar click en el icono de un lápiz alado de la opción.

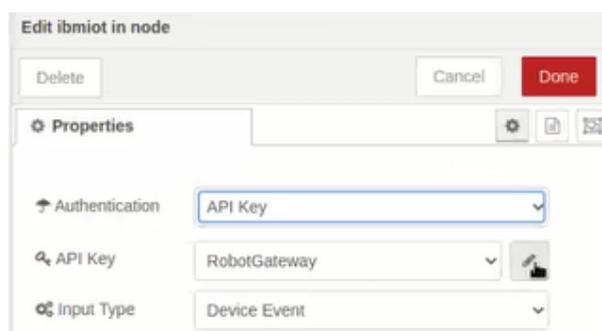


Figura 79. Configuración de nodo IBM IoT con la API Key.

Ingresado en la configuración de la Clave API se va a dar las opciones para poner el nombre, clave, token y nombre del servidor. En nombre, clave y token se transcriben estas que ya fueron creadas con los pasos que resolvió como se puede apreciar en la

figura 25. Luego se procede a seleccionar actualizar (update).

Figura 80. Configuración del nodo IBM IoT (selección del token y clave).

Volvemos a nuestra configuración principal del nodo y en el tipo de dispositivo se procede a escoger arduino ya que se va a utilizar dicho microprocesador, en nombre del dispositivo: robot como se procedió a escribir en el dispositivo en IBM Watson que se aprecia en la figura 21. Y en formato por default en json y se selecciona la opción hecho (done).

Figura 81. Configuración del nodo IBM IoT (selección del tipo de dispositivo y formato)

Se procederá a depurar la información de nuestro nodo IBM IoT la cual se va a utilizar otros nodos, para esto utilizaremos un nodo condicional switch, que se encuentra en la paleta de filtro de nodos y que sirve para saber si el valor no es nulo sino va a recibir o pasar por el canal, hecho esto se procede a seleccionar click en done.

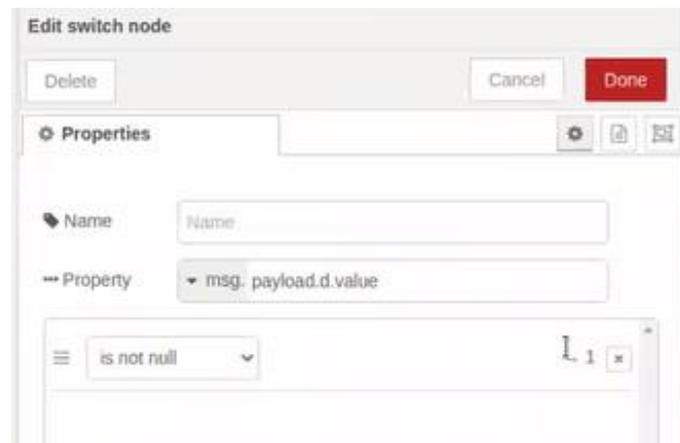


Figura 82. Configuración del nodo switch

Se seleccionara un nodo de depuración más, en este caso es el nodo de funciones que nos servirá para transformar el archivo string formato json a un archivo string formato msg.payload.

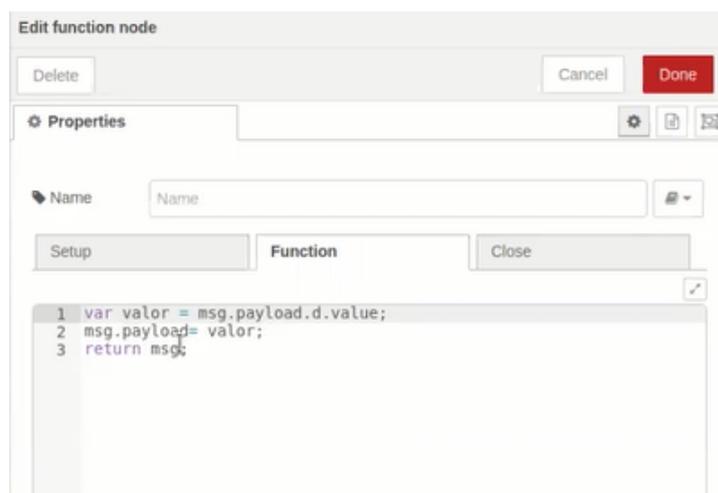


Figura 83. Configuración del nodo de funciones.

Se procederá a usar el nodo serial out lo cual sirve para la comunicación serial del Arduino al Raspberry, ósea comunicación del node red local (Raspberry) al microprocesador (Arduino) que estará ubicado en la paleta en la sección de networks



Figura 84. Escogiendo nodo de comunicación serial.

Damos doble click para entrar a la configuración del nodo para entrar a las opciones.

En este caso vamos a editar el puerto serial y se seleccionara el lápiz que esta alado de esa opción.



Figura 85. Editor del nodo de salida de serial.

Se procede a editar el nodo, la cual tendrá algunas opciones pero solo se procederá a transcribir el puerto en el que está ubicado el Arduino, para ello se recomienda en el Raspberry instalar el Id de Arduino (Programa ejecutable), para que cuando se conecte el Arduino a través de la Id poder reconocer en qué puerto del Raspberry está

conectado.

Concluido en la opción serial port transcribiremos nuestro puerto y en configuraciones en baudios que es la transmisión de señal se escoge 9600, en datos de bits 8, y en bits de retención; el resto por default.

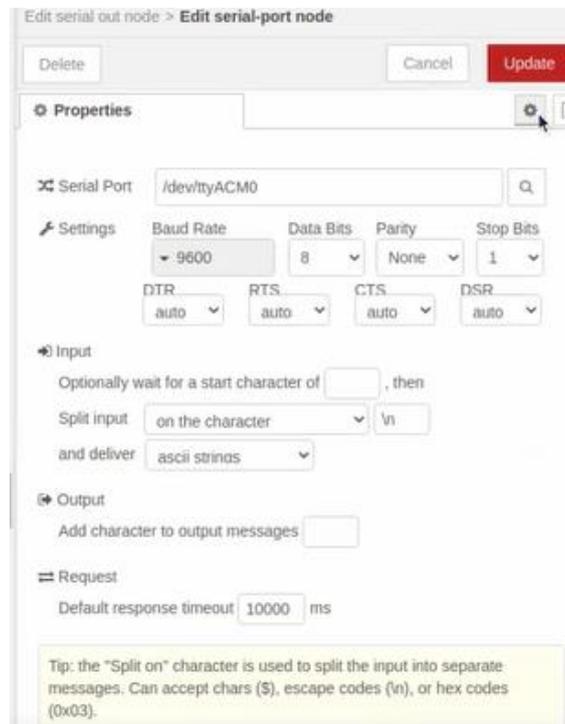


Figura 86. Configurando el número de bit y numero de puerto en el nodo serial

Hecho esto se selecciona update a la configuración del nodo puerto serial y al Node red se procede a dar click en deploy, si todo está en orden se estaría conectado correctamente.

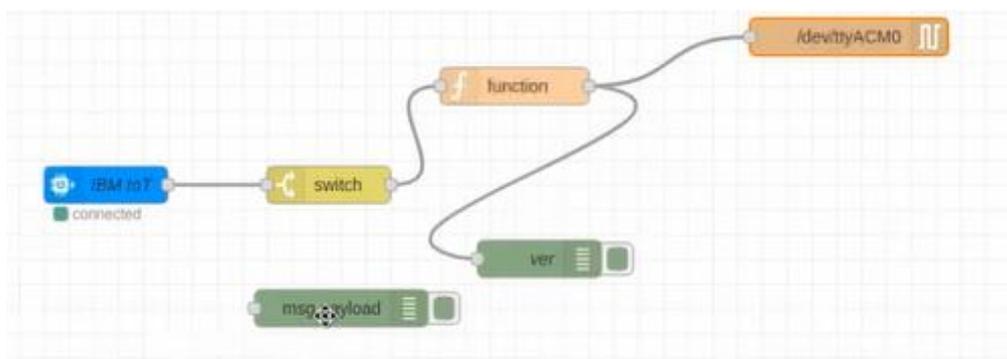


Figura 87. Esquema del node red de la configuración del nodo IBM IoT

3.12. ARDUINO ID

Referente al código de arduino que se va a utilizar, primero es declarar que tipo de variables se va a utilizar en el setup, que son para la utilización de drivers, sensores y cómo se va a trabajar la modalidad de caracteres, se procederá a escoger tipo char para los caracteres y otra variable tipo int para lo que es la velocidad

.

Luego de esto, se va a declarar los pines con su respectivo tipo que se va a utilizar en el microprocesador; como es el caso de los motores son: delantero derecho, delantero izquierdo, trasero derecho y trasero izquierdo; todos estos pines tipo enteros son para salidas tipos digitales.

Para conectar los drivers para los motores, primero hay que tener en cuenta que estos trabajan bajo cuatro parámetro que son: IN1A, IN1B, enable y PWM lo cual se conectara a los pines PWM del arduino y son variables tipo enteros.

Se procedera también a declarar las pines para la bomba, niveles del tanque, paro de emergencia, luces del reservorio y luz piloto estos serán tipo entero.

```
codigo_carrito Arduino 1.8.6
Archivo Editar Programa Herramientas Ayuda
codigo_carrito
char situacion = 0;
int vel;

//Motor delantero Derecha
int motor_ad_dere_1=22;
int motor_ad_dere_2=23;
int motor_ad_dere_enam=24;

//Motor delantero izquierda
int motor_ad_izq_1=25;
int motor_ad_izq_2=26;
int motor_ad_izq_enam=27;
//Motor Tracero Derecha
int motor_atr_dere_1=28;
int motor_atr_dere_2=29;
int motor_atr_dere_enam=30;
//Motor Tracero Izquierda
int motor_atr_izq_1=31;
int motor_atr_izq_2=32;
int motor_atr_izq_enam=33;
|
//PWM DE LOS MOTORES
int pwm_mot_ad_dere=2;
int pwm_mot_ad_izq=3;
int pwm_mot_atr_dere=4;
int pwm_mot_atr_izq=5;

//Bomba
int bomba a=40;
```

Figura 88. Declaración de las variables e arduino ID

Luego, habilitamos el puerto serial, ya que vamos a usar comunicación serial del Raspberry al Arduino, para esto lo habilitamos usando el código: `Serial.begin(9600)` la cual esto va a decir que se va a habilitar el puerto a 9600 baudios.

Luego de crear las variables para el uso de los sensores y drivers, se le asigna a las variables si son salidas o entradas en el microprocesador, en este caso todas son salidas digitales a excepción de los sensores.

En el void loop, se procede a poner las condiciones al programar, en este caso se utilizara una condicional if para que el puerto serial habilitado sea mayor que 0 y la variable situación toma el valor del puerto serial; con esta condición se comienza a programar métodos para los movimientos arriba, abajo, izquierda, derecha, stop, fumigar y emergencia en la cual a cada movimiento se le asigna un tipo de letra.

Comenzaremos a programar las velocidades de los motores la cual se utilizara métodos de movimiento, para eso se utilizara un lazo for diciendo que las velocidad va a ser 0 y va a aumentar de manera paulatina hasta que sea menor que 200 con un tiempo de espera de 100 ms, la cual lo mismo se aplica con los movimiento arriba, abajo, izquierda, derecha y en el caso de los demás actuadores se les asigna una velocidad fija con un tiempo de espera.

```
if(Serial.available()>0){
  situacion=Serial.read();
}
if(situacion == 'a')forward();
if(situacion == 'b')back();
if(situacion == 'c')left();
if(situacion == 'd')right();
if(situacion == 'e')stop_1();
if(situacion == 'f')fumigate();
if(situacion == 'g')emergency();
}

void forward()
{
  for(vel=0;vel<200;vel++){
    m1ad(vel);
    m2ad(vel);
    m3ad(vel);
    m4ad(vel);
    delay(100);
  }
}

void back()
{
  for(vel=0;vel<200;vel++){
    m1at(vel);
```

Figura 89. Asignación de letras a nuestros movimientos y velocidades

Finalmente se procede a crear otros métodos para direccionamiento de los motores, el método es para hacer los sentidos horarios y anti horarios de los motores utilizando lógica de altos y bajos, esta programación se aplica con cada uno de los motores, en este caso como son 5 motores incluyendo el de la bomba se aplica el mismo método para cada uno y con eso concluye la programación en nuestro arduino

```
Archivo Editar Programa Herramientas Ayuda
codigo_carrito $
}
void m1ad(int oenam1){
    digitalWrite(motor_ad_dere_1,HIGH);
    digitalWrite(motor_ad_dere_2,LOW);
    digitalWrite(motor_ad_dere_enam,HIGH);
    analogWrite(pwm_mot_ad_dere,oenam1);
}
void m1at(int oenam1t){
    digitalWrite(motor_ad_dere_1,LOW);
    digitalWrite(motor_ad_dere_2,HIGH);
    digitalWrite(motor_ad_dere_enam,HIGH);
    //analogWrite(pwm_mot_ad_dere,oenam1t);
}
void m2ad(int oenam2){
    digitalWrite(motor_ad_izq_1,HIGH);
    digitalWrite(motor_ad_izq_2,LOW);
    digitalWrite(motor_ad_izq_enam,HIGH);
    analogWrite(pwm_mot_ad_izq,oenam2);
}
void m2at(int oenam2t){
    digitalWrite(motor_ad_izq_1,LOW);
    digitalWrite(motor_ad_izq_2,HIGH);
    digitalWrite(motor_ad_izq_enam,HIGH);
    analogWrite(pwm_mot_ad_izq,oenam2t);
}
void m3ad(int oenam3){
    digitalWrite(motor_atr_dere_1,HIGH);
    digitalWrite(motor_atr_dere_2,LOW);
    digitalWrite(motor_atr_dere_enam,HIGH);
}
```

Figura 90. Asignación de métodos para el giro de los motores

3.13. PROTEUS

En la creación de las placas, primero se realiza el diseño esquemático electrónico con los componentes utilizados para hacer la placa, lo primero es buscar los componentes electrónicos con sus respectivos valores y códigos, lo conectamos entre sí para obtener un circuito cerrado la cual también se conecta su respectiva tierra y alimentación común de los componentes, en el caso de la creación de la placa de arduino se procedió a crear un nuevo componente y hacer las medidas desde cero ya que la librería de arduino mega no existe en Proteus.

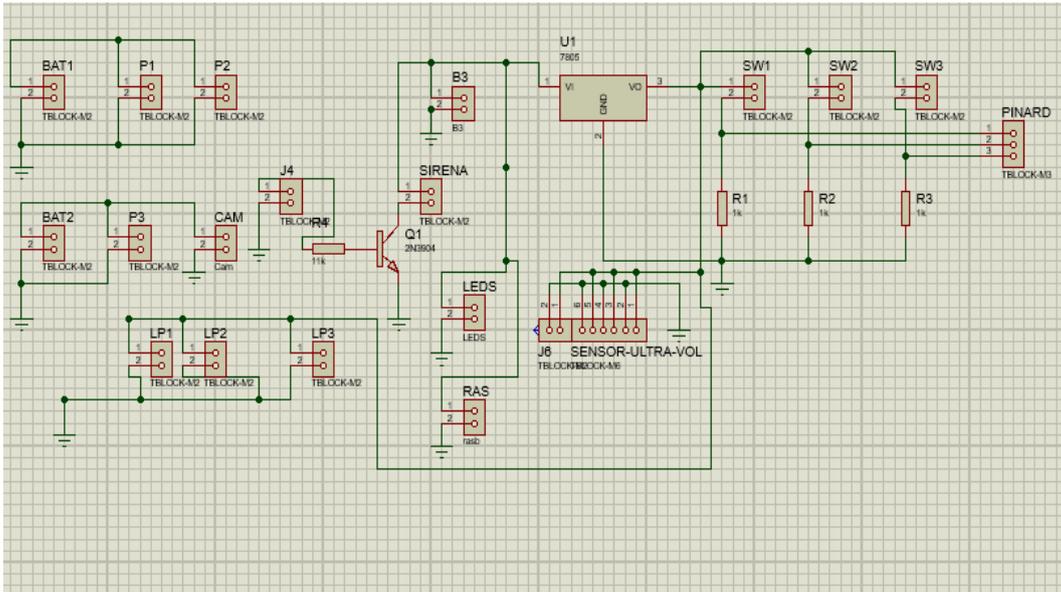


Figura 91. Esquema electrónico en Proteus

Como siguiente paso, abrimos nuestro Ares de Proteus para la creación de la placa PCB, la cual como ya se procedió a realizar el esquema electrónico de los componentes, nos va a salir ya en Ares lo ya utilizado, solo hay que unirlos y distribuirlos para que las pistas no choquen entre si y proceder a escoger de cuantas capas va a ser el PCB, en este caso solo se escogerá de una y se envía a una empresa a que imprima la placa.

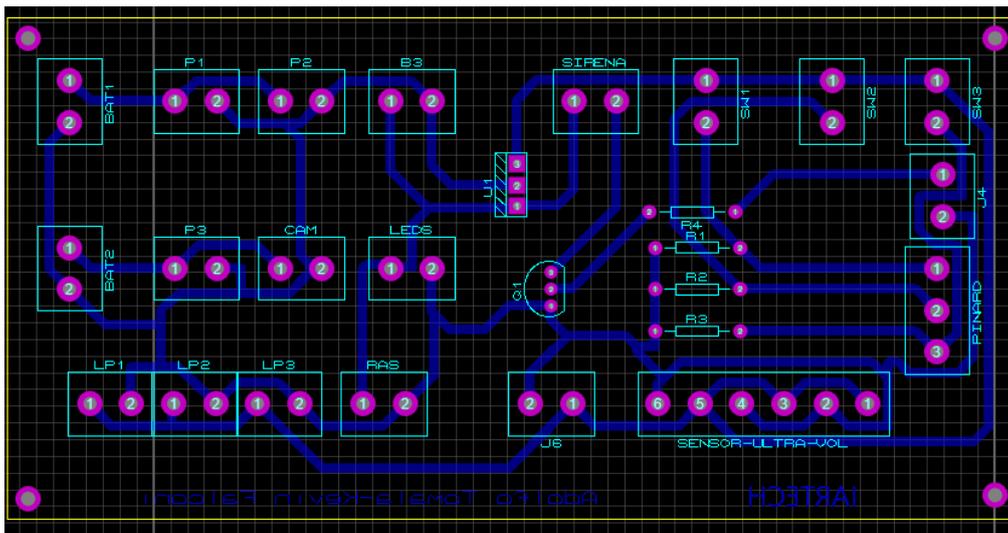


Figura 92. Diseño de PCB en Proteus

3.14. AUTOCAD

En la creación de la estructura de robot móvil diferencial, se utilizó un prototipo propio con planos sólidos en 3D, para saber las medidas y escalas la cual se hizo por formas separadas y al final se crea solo un cuerpo.

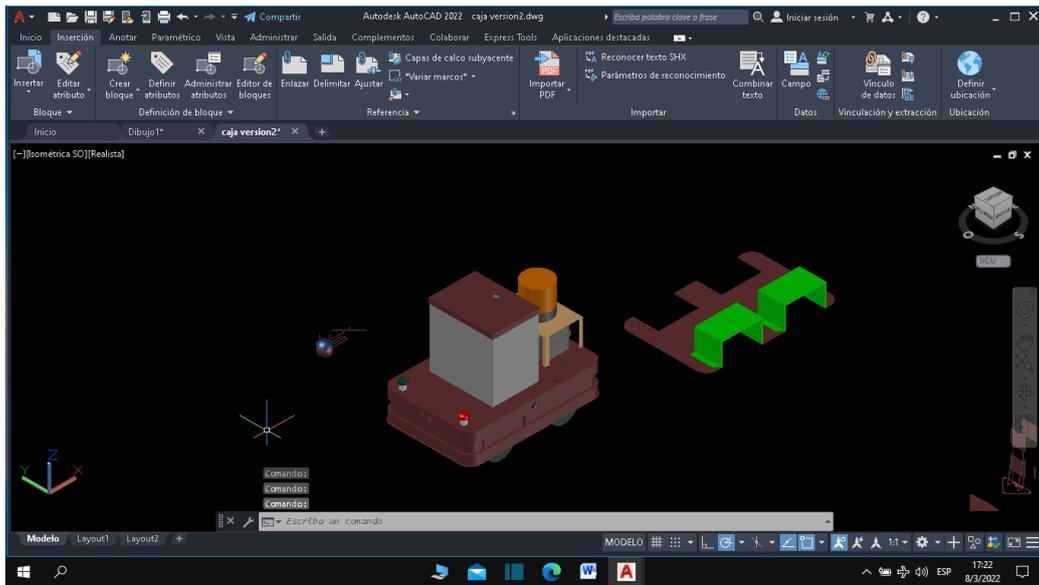


Figura 93. Diseño de planos en Autocad

4. RESULTADOS

- Se preparó un modelo robótico diferencial por telemetría, con comunicación de red pública y privada para practicas o demostraciones del uso de la robótica en la vida diaria, la cual consta con un Raspberri PI 4 diseñado para su uso en las diversas prácticas destinadas a comunicación 4.0 en las asignaturas orientadas a robots móviles que se importan en la Politécnica Salesiana.



- Se ha propuesto modelos de PCB que facilitará la implementación de elementos como sensores, bomba de agua, baliza, controladores, baterías y actuadores.

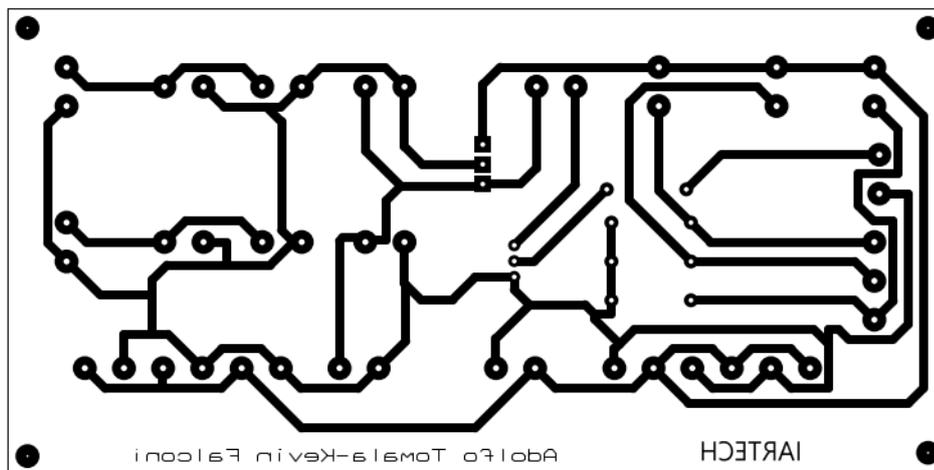


Figura 94. Placa de distribución de sensores y baliza

- Se creó un dashboard en la nube fácil de manejar para el usuario con protocolos de seguridad

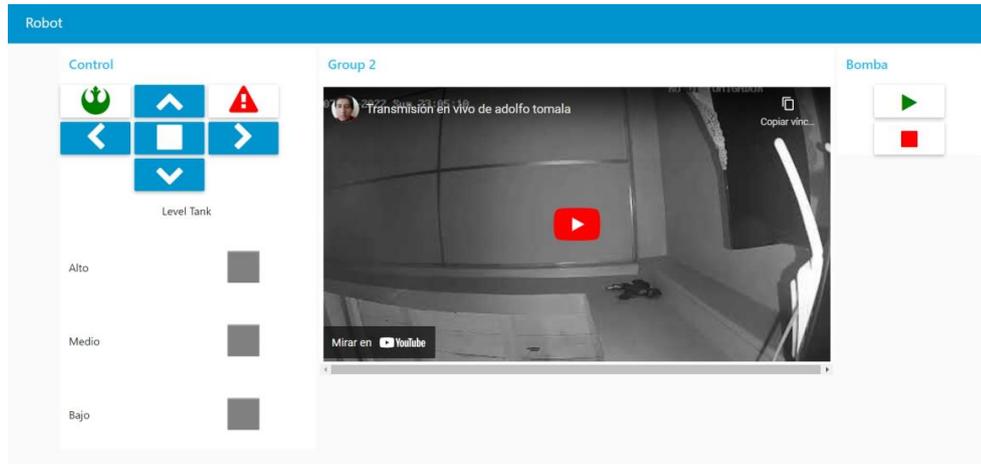


Figura 95. Tablero o dashboard

- Se logró la comunicación del Raspberry a la nube mediante softwares IBM cloud y Microsoft Azure para el manejo del prototipo dando como maestro el Raspberry y esclavo Arduino Mega para el manejo de sus actuadores y componentes.



Figura 96. Transmisión de la cámara IP a YouTube

- Se logró vincular la cámara IP al dashboard mediante software OBS Studio y YouTube

- Gracias a las pruebas realizadas se logró un control óptimo del movimiento, fumigación e indicadores del robot mediante el node red configurado en el Raspberry
- Se entrega el equipo un robot móvil diferencial aspersor de alcohol con telemetría completamente funcional, con accesorios para el prototipo como cargador y dashboard para el uso en el aula de la Universidad Salesiana.

4.1. ANÁLISIS DE RESULTADOS

Por medio de resultados estudiados a través de una encuesta realizada explicando funcionamiento y diferentes aplicaciones se llegó a la siguiente tabla de resultados.

Tabla 4. Diferencias Hombre VS Maquina en sistemas de fumigación

Ventajas y desventajas hombre vs maquina en el área de fumigación				
	Hombre		Maquina	
	Si	No	Si	No
Riesgo a contagio en una zona infectada	X			X
Trabajos sin descansos		x	x	
Trabajos en lugares de alto riesgo		x	x	
Adaptables a cualquier ambiente	X		x	
Facilidad de transporte	X		x	
Riesgos de propagación una carga viral	X			X
Auto implementación de nuevas tecnologías	X		x	

Po medio de esta tabla se llega al resultado más importante, que la fumigación por medio de maquina móvil programable casos de altos riesgos, es más conveniente su uso para evitar pérdidas humanas o la propagación de una pandemia o epidemia.

5. CONCLUSIONES

- Para el desarrollo del proyecto, se usaron diferentes software como Arduino ID, node red y su herramienta de enlace IBM Cloud, la cual se puede configurar la comunicación y la adquisición de los sensores y actuadores para el control del robot diferencial de manera autónoma cumpliendo con lo establecido en el proyecto.
- Se desarrolló un aplicativo virtual (dashboard), que por medio de node red con una conexión de una IP pública o privada, se puede utilizar para el control autónomo del robot diferencial para fumigar o rociar el producto químico en una zona de alto contagio.
- Se implementó mediante node red por puertos de enlaces publicas la adquisición de una cámara IP, para que se logre el análisis de la dirección del robot por medio de video y vincularlo a nuestro dashboard
- Se obtuvo una comunicación por medio de la nube el control de los actuadores, sensores y demás componentes del prototipo
- Se logró comunicar de manera eficiente el robot tanto como a una red privada como a una red publica

6. RECOMENDACIONES

- Revisar el porcentaje de carga de las baterías antes de iniciar el proceso de arranque del robot
- El prototipo diseñado tiene funciones de control robustas que permiten agregar sistemas de control más complejos en futuras actualizaciones
- Implementar controladores y mejores sensores para analizar y controlar mejor la trayectoria
- El router se puede configurar para que el robot se puede manejar a través de una red pública y privada para eso se debe configurar el router y el Raspberry.
- Durante el encendido se recomienda que el robot este en un lugar estable y segura antes de iniciar el proceso.
- Es recomendable mantener al robot en un lugar fresco y ventilado a fin que sus componentes electrónicos y baterías estén refrigeradas.
- Se recomienda tener una buena conexión y comunicación a internet superior a 40 MB relación 2 a 1

7. REFERENCIAS BIBLIOGRÁFICAS

Robot móvil concepto:

- [1] <https://www.ixograma.com/tecnologia/robots-moviles-definicion-funcionamiento-y-aplicacion/>

Ruedas con tracción diferencial:

- [2] http://platea.pntic.mec.es/vgonzale/cyr_0204/cyr_01/robotica/movil.htm

Raspberry Pi concepto:

- [3] <https://www.xataka.com/makers/cero-maker-todo-necesario-para-empezar-raspberry-pi#:~:text=La%20Raspberry%20Pi%20es%20la,bajo%20nivel%2C%20reloj...>

Arduino concepto:

- [4] <https://arduino.cl/que-es-arduino/>

Node red concepto:

- [5] [https://www.ain.es/formacion/curso/curso-node-red-developer/#:~:text=Node%2DRed%20es%20una%20herramienta,de%20las%20cosas%20\(IoT\).](https://www.ain.es/formacion/curso/curso-node-red-developer/#:~:text=Node%2DRed%20es%20una%20herramienta,de%20las%20cosas%20(IoT).)

IBM Cloud concepto:

- [6] <https://cloud.ibm.com/docs/overview?topic=overview-what-is-platform&locale=es>

IoT concepto:

- [7] <https://www.sap.com/latinamerica/insights/what-is-iot-internet-of-things.html>

Concepto cliente/servidor:

- [8] <https://definicion.de/cliente-servidor/>

Concepto dashboard:

- [9] <https://www.cyberclick.es/numerical-blog/que-es-un-dashboard>

Concepto desinfección:

- [10] <https://www.iblspecific.com/es/la-desinfection-definition/>

Concepto de telemetría:

- [11] <https://telemetrik.co/telemetria-que-es-aplicaciones-casos-de-uso/>

Sensor ultrasónico concepto:

- [12] <https://www3.gobiernodecanarias.org/medusa/ecoblog/fsancac/2018/02/06/arduino-sensor-ultrasonico-hc-sr04/>

Driver Concepto:

- [13] <https://www.yoreparo.com/es/industria/electronica/preguntas/421399/definicion-de-driver>

Concepto de carga bacteriana:

- [14] <https://esferasalud.com/covid-19/carga-microbiana>

Node.js:

- [15] <https://openwebinars.net/blog/que-es-nodejs/#:~:text=Node.,un%20programa%20escrito%20en%20JavaScript.>

Achivo.js:

- [16] <https://www.nextu.com/blog/que-es-json/>

Industria 4.0:

- [17] <https://www.epicor.com/es-mx/resources/articles/what-is-industry-4-0/#:~:text=Industria%204.0%20se%20refiere%20a,los%20datos%20en%20tiempo%20real.>

IBM vs competidores:

- [18] <https://www.ibm.com/cloud/info/ibm-cloud-vs-competition>

Bare metal concepto:

- [19] <https://www.ionos.es/digitalguide/servidores/know-how/servidores-bare-metal-definicion-y-configuracion/>

Prototipo de un robot móvil:

- [20] <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwikx-bb2rX2AhVwRDABHa4aC4UQFnoECAMQAQ&url=https%3A%2F%2Ftesis.i>

pn.mx%2Fjspui%2Fbitstream%2F123456789%2F18900%2F1%2FPrototipo
%2520de%2520robot%2520m%25C3%25B3vil%2520terrestre%2520de%25
20fumigaci%25C3%25B3n%2520autom%25C3%25A1tica%2520para%2520
hortalizas.pdf&usg=AOvVaw37qVTaxnUmQUckPflE9oDN

Porcentaje actual de los contagiados:

[21] <https://es.statista.com/grafico/24335/vacunacion-contra-el-coronavirus-america-latina/>

Diferencia entre pandemia y endemia:

[22] <https://observatorio.medicina.uc.cl/epidemia-pandemia-o-endemia/>

Coronavirus:

[23] https://www.who.int/es/health-topics/coronavirus#tab=tab_1

OMS declara pandemia SARS 2:

[24] https://www3.paho.org/hq/index.php?option=com_content&view=article&id=15756:who-characterizes-covid-19-as-a-pandemic&Itemid=1926&lang=es

Variación de casos covid:

[25] <https://www.salud.gob.ec/variacion-en-casos-por-reclasificacion/>

Cámara IP:

[26] <https://www.redatel.net/html/camara-ip-conceptos-basicos.html>

Microsoft Azure:

[27] https://azure.microsoft.com/es-mx/free/search/?&ef_id=EAlalQobChMlt9LOz6mO-gIVmG1vBB22kgAnEAAYASAAEgl2VfD_BwE:G:s&OCID=AIDcmmvcssag76_SEM_EAlalQobChMlt9LOz6mO-gIVmG1vBB22kgAnEAAYASAAEgl2VfD_BwE:G:s&gclid=EAlalQobChMlt9LOz6mO-gIVmG1vBB22kgAnEAAYASAAEgl2VfD_BwE

OBS Studio:

[28] <https://www.mancomun.gal/es/solucion-tic/obs-studio/#:~:text=OBS%20Studio%2C%20u%20Open%20Broadcaster,profesionales%20y%20f%C3%A1cil%20de%20utilizar.>

Motor reductor especificaciones:

[29] <https://www.pololu.com/product/4741>

Raspberry PI especificaciones:

[30] https://octopart.com/raspberry+pi+4+8gb+basic+kit+uni+version-raspberry+pi-119101736?gclid=EAlaIqobChMIrMPHjciT-gIVCMOGCh2A8AlfEAAYASAAEgIhDfD_BwE

Driver motor especificaciones:

[31] <https://www.pololu.com/product/2507>

Bomba de agua especificaciones:

[32] https://www.amazon.com/stores/page/1541C781-DFC8-4771-85CB-C92689E0C8AC/?_encoding=UTF8&store_ref=SB_A07676762OHIPC1LBP3FA&pd_rd_plhdr=t&aaxitk=fc1db4ca187b5502d3d9055dbf13bf14&hsa_cr_id=3836373570001&lp_asins=B074MZYS37%2CB01N75ZIXF%2CB07FT87XJY&lp_query=12v%20diaphragm%20pump&lp_slot=desktop-hsa-3psl&ref_=sbx_be_s_3psl_mbd_hl&pd_rd_w=R0HOq&content-id=amzn1.sym.85014337-b4b1-4f3c-85f8-828d3b814280%3Aamzn1.sym.85014337-b4b1-4f3c-85f8-828d3b814280&pf_rd_p=85014337-b4b1-4f3c-85f8-828d3b814280&pf_rd_r=WS4Y1D3EDB0MG6VR71R3&pd_rd_wg=uW8T8&pd_rd_r=a3d3116e-0632-4f14-b457-99c0379c6cc7

Batería especificaciones:

[33] https://hobbyking.com/en_us/turnigy-graphene-6000mah-6s-75c-lipo-pack-w-xt90.html?gclid=EAlaIqobChMIwryz9MqT-gIVmpSGCh2QRQMBEAYASAAEgIRI_D_BwE&gclid=EAlaIqobChMIwryz9MqT-gIVmpSGCh2QRQMBEAYASAAEgIRI_D_BwE

Regulador de voltaje especificaciones:

[34] https://www.amazon.com/HiLetgo-Step-down-Converter-1-25-37V-Voltmeter/dp/B00LSEBYHU/ref=sr_1_1_1_sspa?keywords=Adjustable+Voltage+Regulator&qid=1663134679&sr=8-1-spons&psc=1&spLa=ZW5jcnlwdGVkUXVhbGlmaWVyPUEzM1pNQktTSzYyUDYxJmVuY3J5cHRIZEikPUEwNTA3NjUwMU1STDRCT1JKOURVTiZlbnN

yeXB0ZWRBZEIkPUEwMzU2MjEyMTdKukZSRUuzU1BOSyZ3aWRnZXRO
YW1IPXNwX2F0ZiZhY3Rpb249Y2xpY2tSZWRpcmVjdCZkb05vdExvZ0Nsa
WNRpXRydWU=

Sensor tipo boya especificaciones:

[35] https://articulo.mercadolibre.com.ec/MEC-518089895-sensor-de-nivel-de-agua-tipo-flotador-switch-electronics-_JM#position=1&search_layout=stack&type=item&tracking_id=6d29e8cae321-4b8d-bddb-462ac5f8efd3

Arduino mega especificaciones:

[36] <https://arduino.cl/arduino-mega-2560/>

Sensor ultrasónico especificaciones:

[37] <https://naylampmechatronics.com/sensores-proximidad/10-sensor-ultrasonido-hc-sr04.html>

Concepto sensor de nivel tipo boya:

[38] <http://www.eicos.com/datos-tecnicos/que-es-un-sensor-de-nivel/#:~:text=Los%20sensores%20de%20nivel%2C%20tambi%C3%A9n,o%20cierra%20el%20contacto%20el%C3%A9ctrico.>

Robot explorador con tecnología pitsco:

[39] <https://dspace.ups.edu.ec/handle/123456789/19379>

Como crear una máquina virtual:

[40] <https://www.youtube.com/watch?v=Vh-o9Jcxh3Y>

Epidemia, pandemia y endemia definición:

[41] <https://observatorio.medicina.uc.cl/epidemia-pandemia-o-endemia/>

Anexos

Proceso de montaje del robot

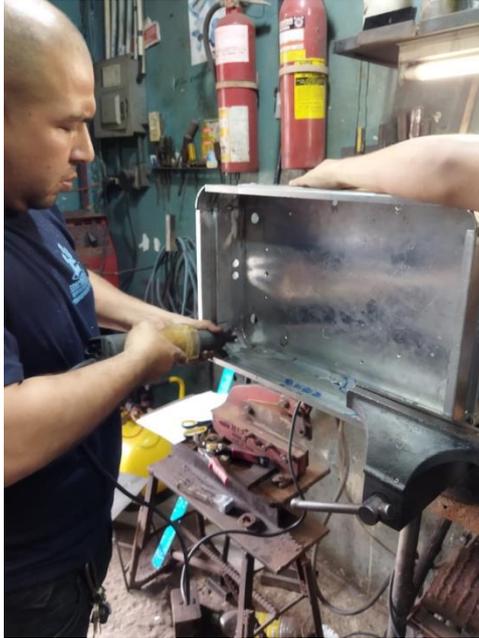


Figura 97. Proceso de dobles y cortada de la estructura



Figura 98. Proceso de limar las imperfecciones de la estructura del robot



Figura 99. Proceso de empaste de la estructura del robot móvil

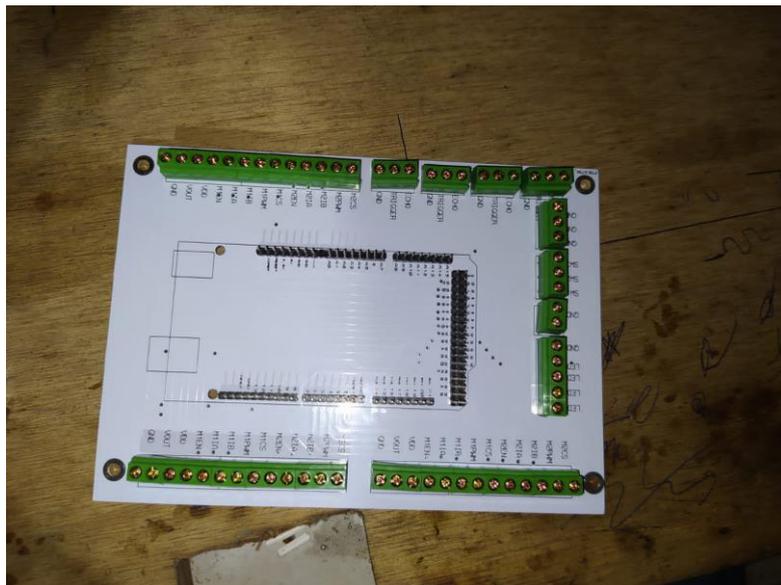


Figura 100. Ensamblaje de las placas del robot fumigador

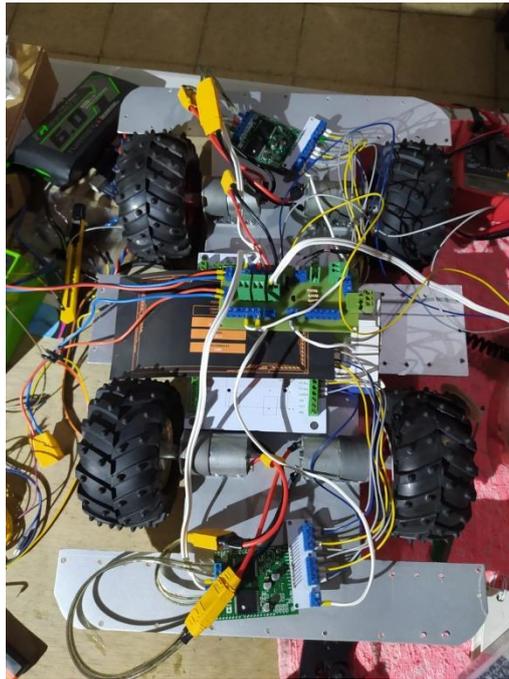


Figura 101. Montaje de actuadores y tarjetas electrónicas a la estructura del robot

Tabla 5. PRESUPUESTO PARA LA CONSTRUCCIÓN DE UN ROBOT MÓVIL DIFERENCIAL ASPERSOR DE ALCOHOL, CON TECNOLOGÍA IOT Y TELEMETRÍA

Materiales	Cantidad	Precio Unitario (dolares)	Precio Total (dolares)
Baterías de grafeno	2	\$116.34	\$232.68
Cámara IP	1	\$70	\$70
Aspersor	1	\$30	\$30
Lámina de aluminio 30x30	1	\$40	\$40
Arduino Mega	1	\$26	\$26
Raspberry Pi4 8GB	1	\$120	\$120
Pololu dual VNH5019 Motor driver	4	\$136	\$544
30.1 metal gear motor 37dx52l	4	\$42.41	\$169.66
Bomba de diafragma de presión de agua con interruptor de presión	1	\$35.62	\$35.62
Conectores hembra XT90	1	\$5.44	\$5.44

Convertidor bidireccional modulo 3.3 a 5v para Arduino	2	\$11.89	\$23.78
Proceso de dobléz y cortado de aluminio	1	\$50	\$50
Baliza	1	\$20	\$20
Sensor ultrasónicos	4	\$3	\$12
Sensor de boya	3	\$4	\$12
Reservorio	1	\$20	\$20
Cables, remaches y caja plasticas, manguera y conectores	20	\$60	\$60
Acrílico reservorio	1	\$120	\$120
Placas electrónicas para drivers	2	\$15	\$30
Placa electronica para Arduino	1	\$50	\$50
Placa electronica leds	1	\$25	\$25
Placa electronica distributiva	1	\$40	\$40
Router	1	\$40	\$40
Proceso de pintada y empastado de la estructura	1	\$150	\$150
Llantas	4	\$15	\$60
Servicio Microsoft Azure mensual	3	\$60	\$180
TOTAL			\$2116.18

MANUAL DE USUARIO

- | | |
|---------------------------------|--|
| 1. Selector de On/Off | 5. Fumigador tipo de riego por goteo |
| 2. Paro de emergencia | 6. Motor de riego 4.51 PM |
| 3. Cámara IP | 7 Baliza o luz indicadora de comienzo de riego |
| 4. Reservorio de 10 L (químico) | |

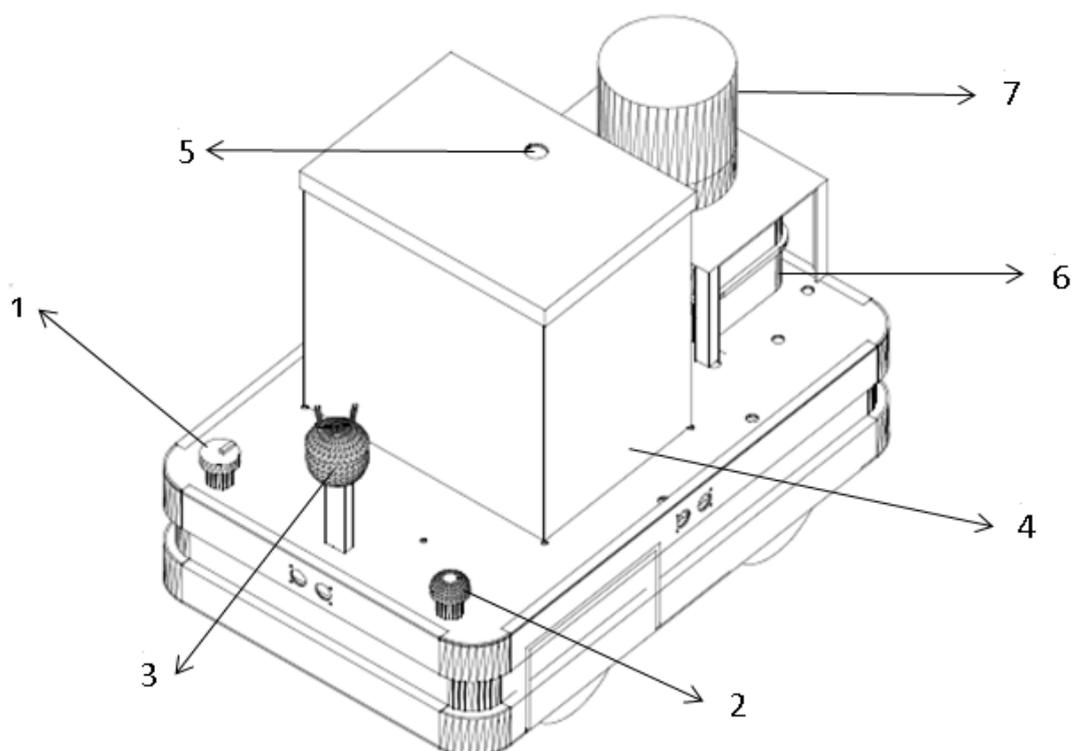


Figura 102. Diseño final del robot fumigador

Pasos para dar inicio el robot:

- 1.- Encender el robot con el selector
- 2.- Llenar el tanque con el químico que se desea fumigar
- 3.- Acceder al siguiente link para el manejo de nuestro tablero o dashboard
<http://20.89.142.188/ui>
- 4.- Escribir el usuario y contraseña en la pagina

Usuario: ROBOT12022

PASSWORD: ELECTRO2022

DASHBOARD:

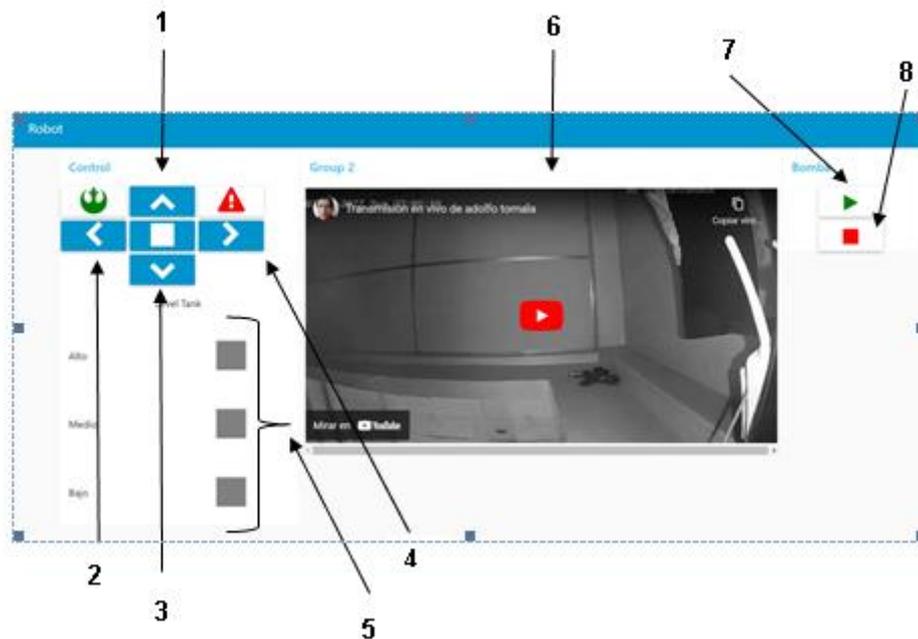


Figura 103. Controles del dashboard

1. Botón de movimiento del robot (adelante)
2. Botón de movimiento del robot (izquierda)
3. Botón de movimiento del robot (abajo)
4. Botón de movimiento del robot (derecha)
5. Indicadores de nivel de agua
6. Ventana de visualización de cámara del robot
7. Botón de encendido de bomba de agua (fumigación cíclica cada segundo con paros de 2 segundos).
8. Botón de apagado de bomba

Pasos para el proceso de fumigación:

- Una vez encendido el robot, nos dirigimos al área contaminada guiándonos por medio de la cámara y con la ayuda de los sensores ultrasónicos
- Ya en el área contaminada, se procede a encender la bomba de agua y se procede por medio del control de movimiento (adelante, abajo, derecha e izquierda) a mover el robot para que fumigue toda el área deseada.
- Terminado la desinfección del área, se procede a detener la bomba de fumigación y sacar el robot del área contaminada.

MANUAL TÉCNICO

PLANOS:

Estructura del robot

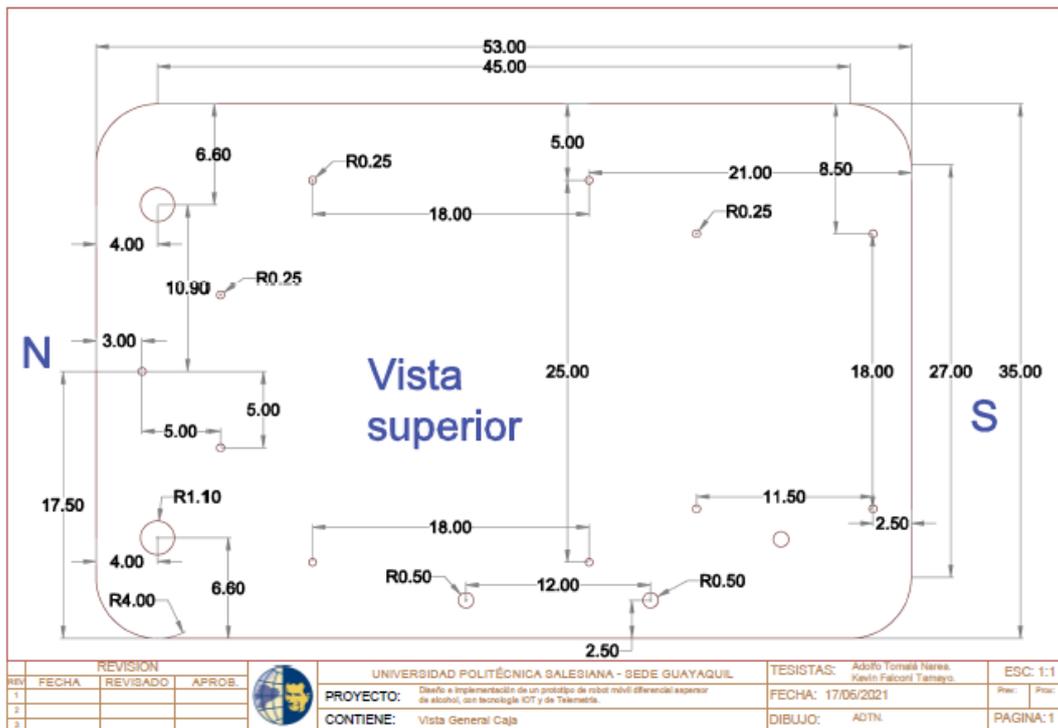


Figura 104. Vista superior de la estructura del robot fumigador

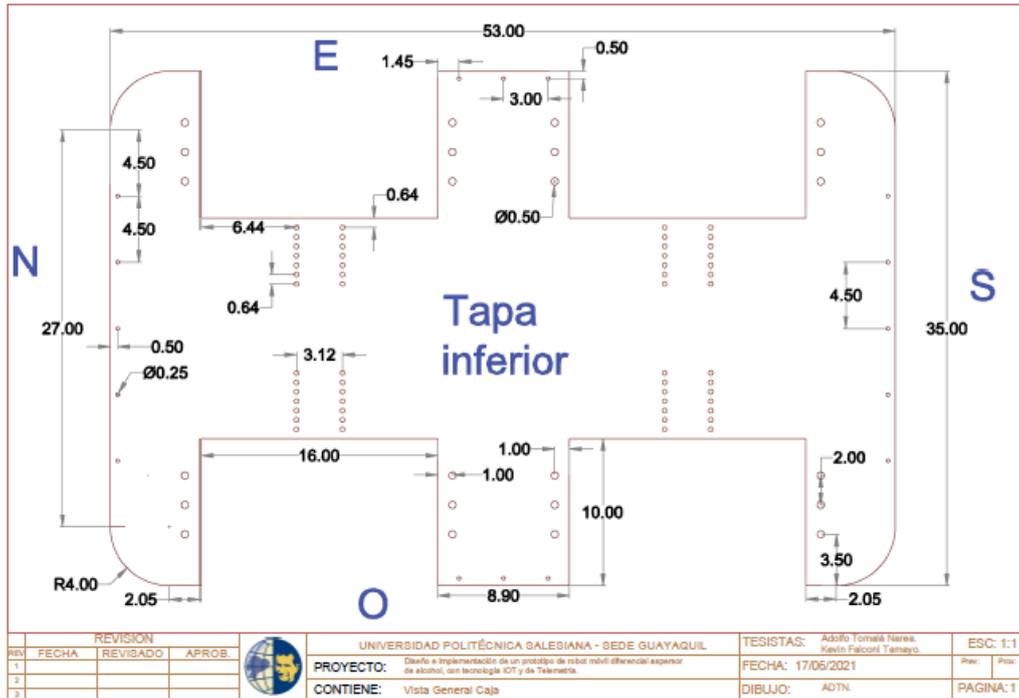


Figura 105. Vista inferior de la estructura del robot fumigador

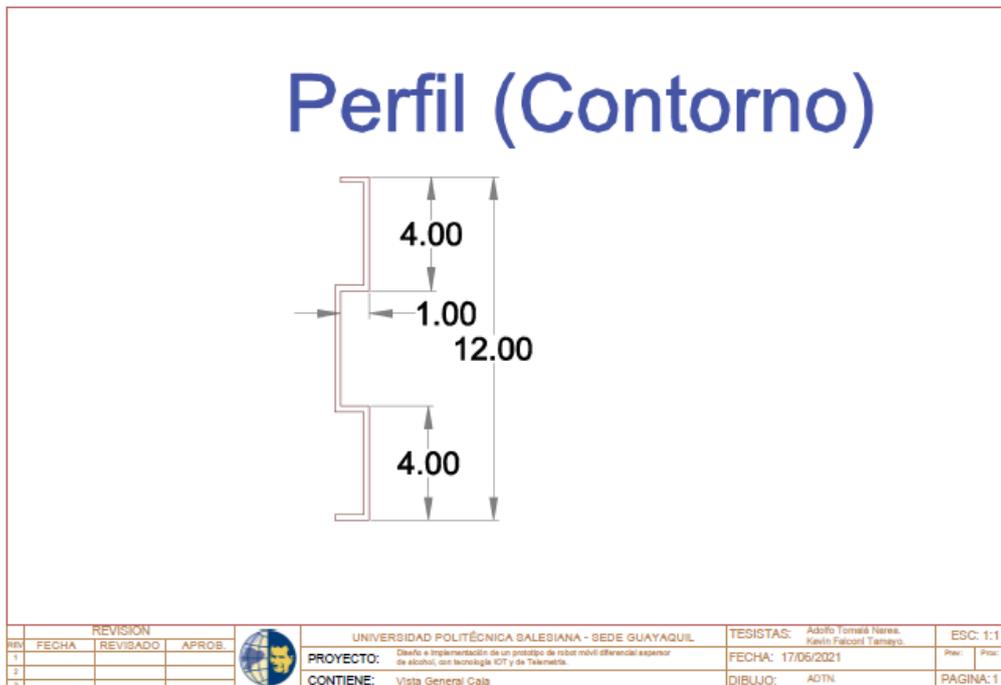


Figura 106. Vista del contorno del robot fumigador

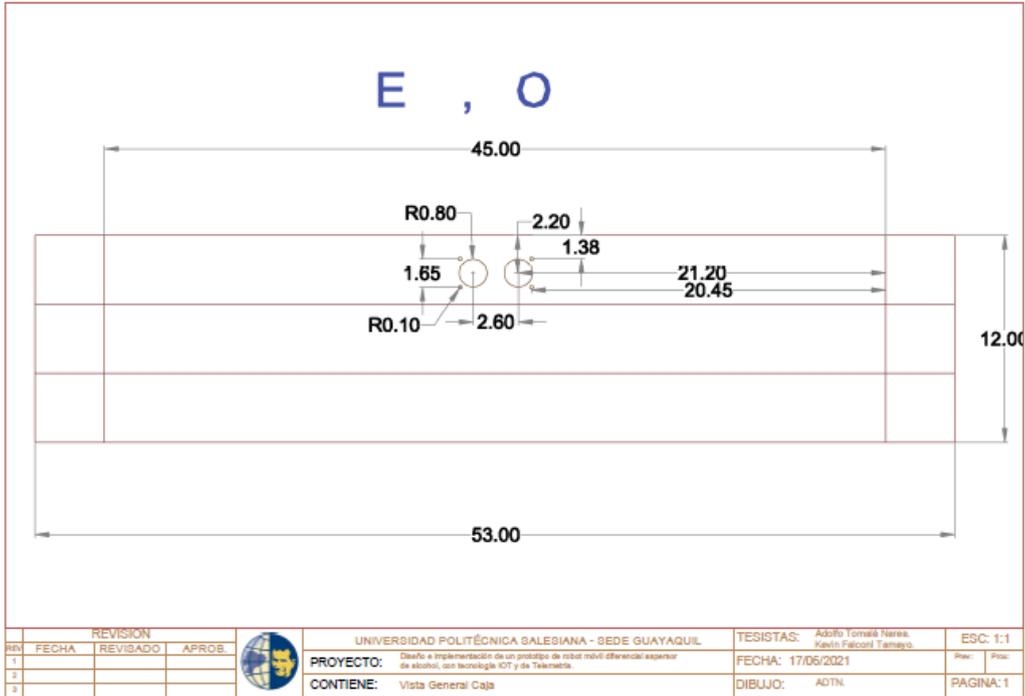


Figura 107. Vista Este y Oeste del robot fumigador

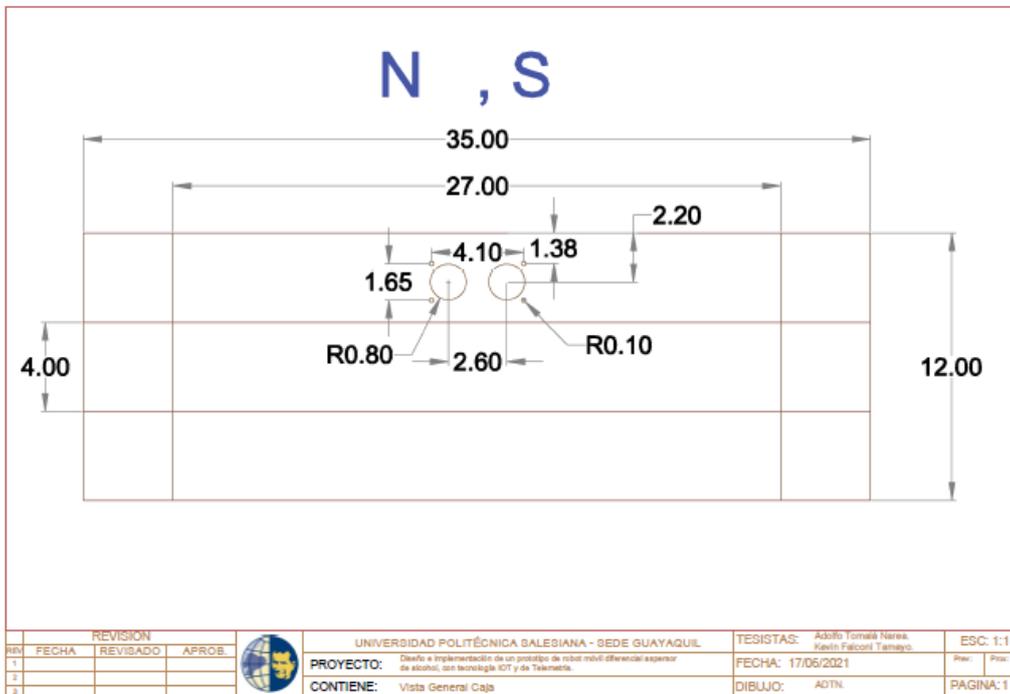


Figura 108. Vista Norte y Sur del robot fumigador

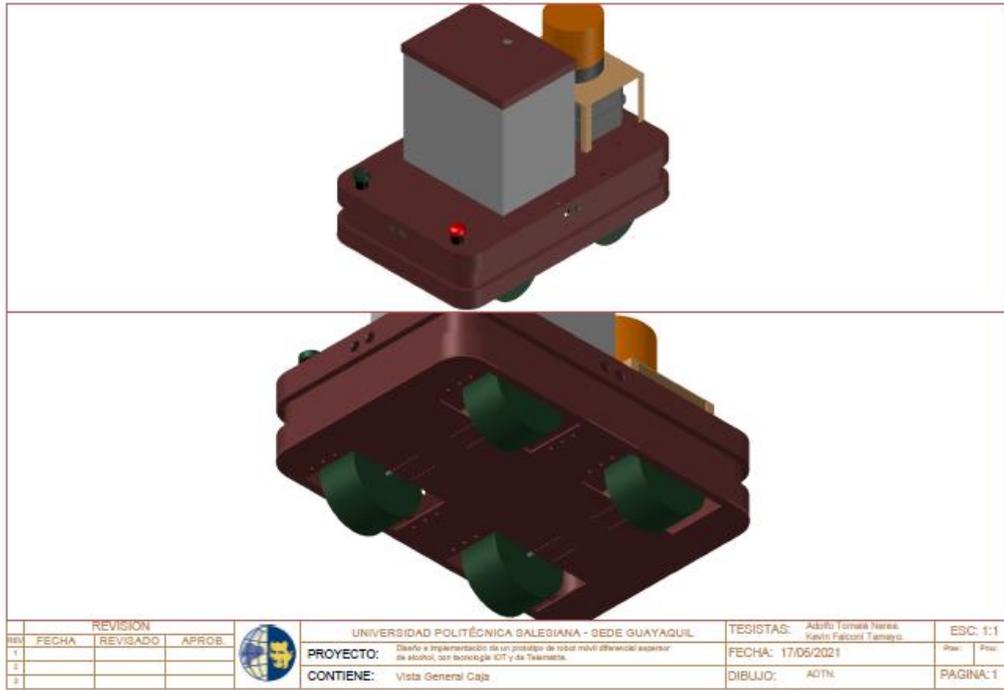


Figura 109. Vista en 3D superior e inferior del robot fumigador

Diseño placa de distribución para Arduino Mega

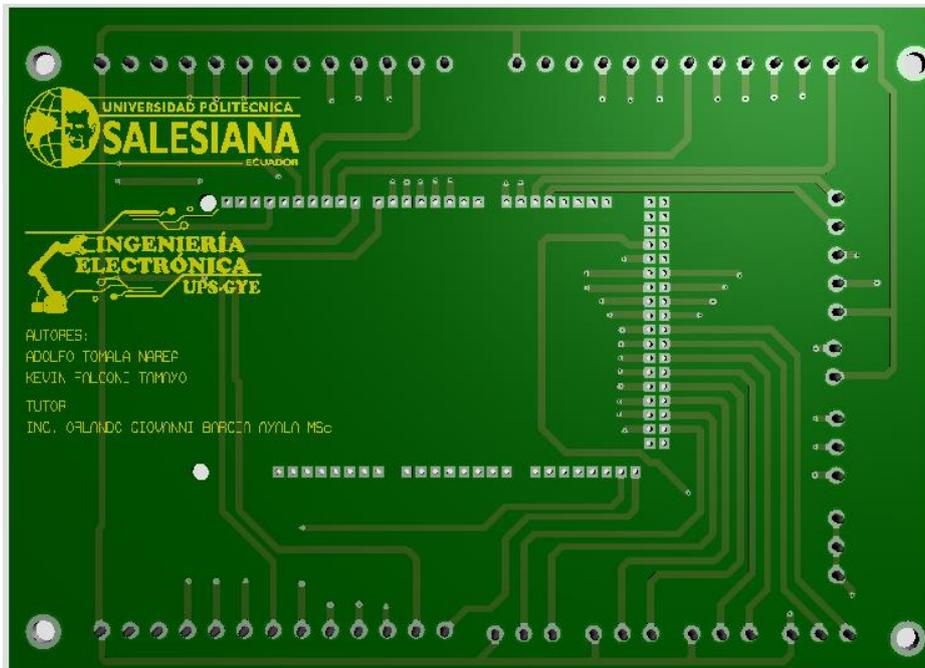


Figura 110. Vista inferior de placa de distribución del Arduino Mega

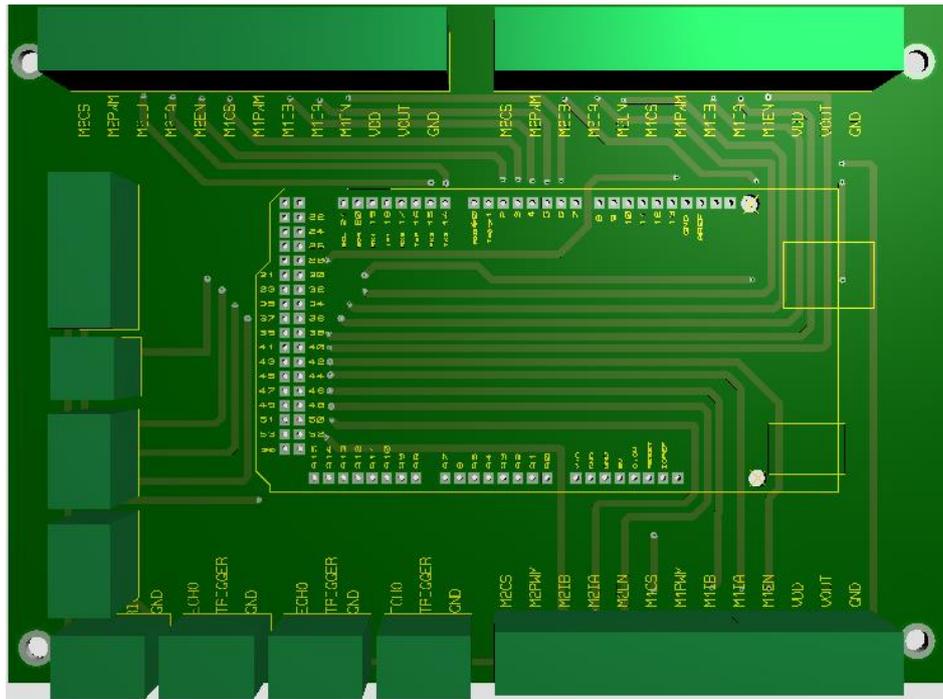


Figura 111. Vista superior de placa de distribución del Arduino Mega

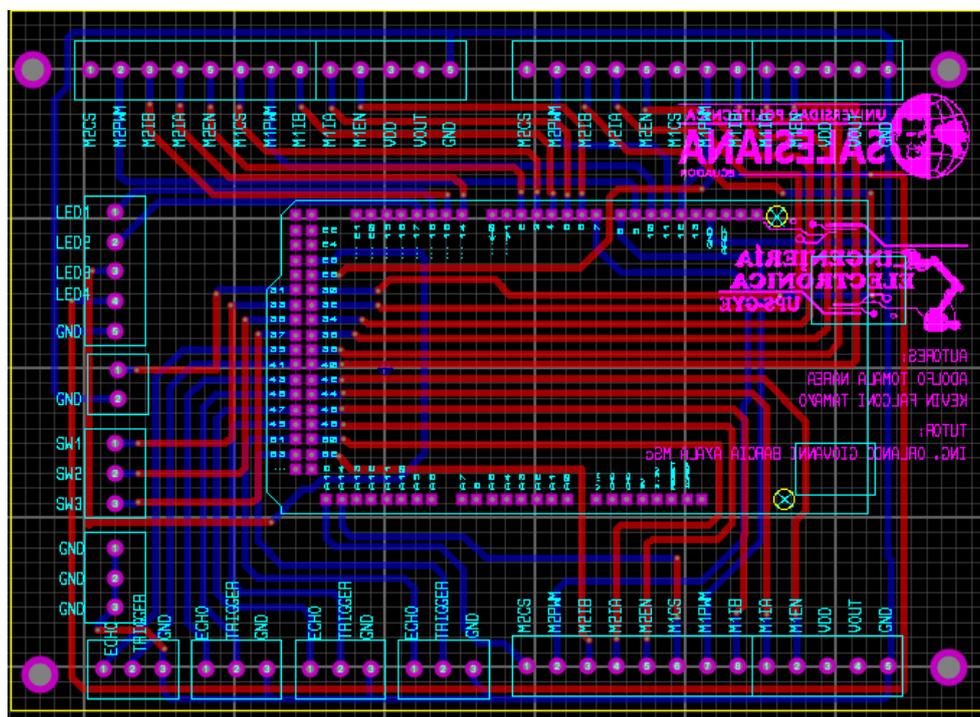


Figura 112. PCB de la placa de distribución del Arduino Mega

Diseño de placa de distribución de indicadores leds

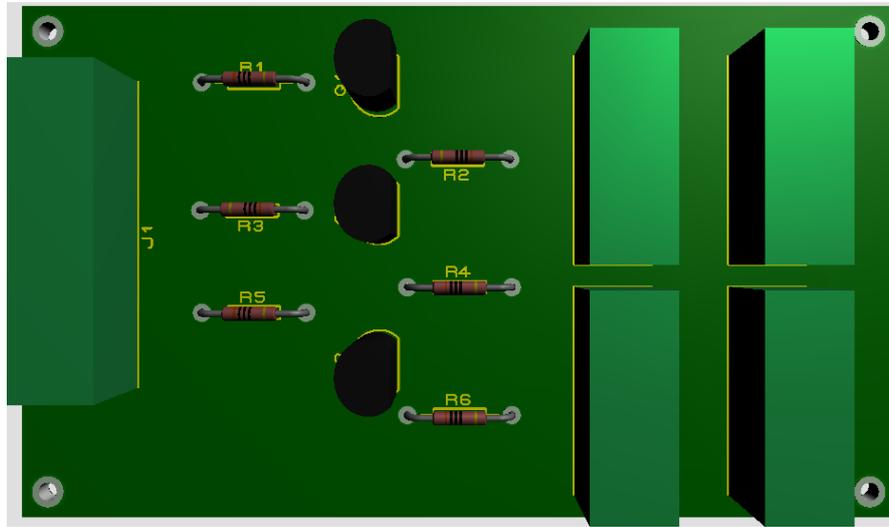


Figura 113. Vista superior de la placa de distribución de indicadores leds.

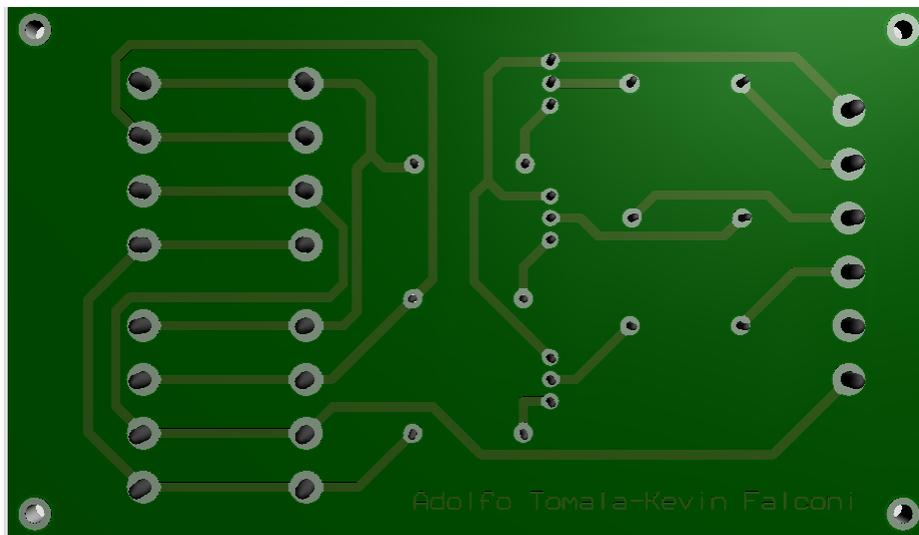


Figura 114. Vista inferior de la placa de distribución de indicadores leds.

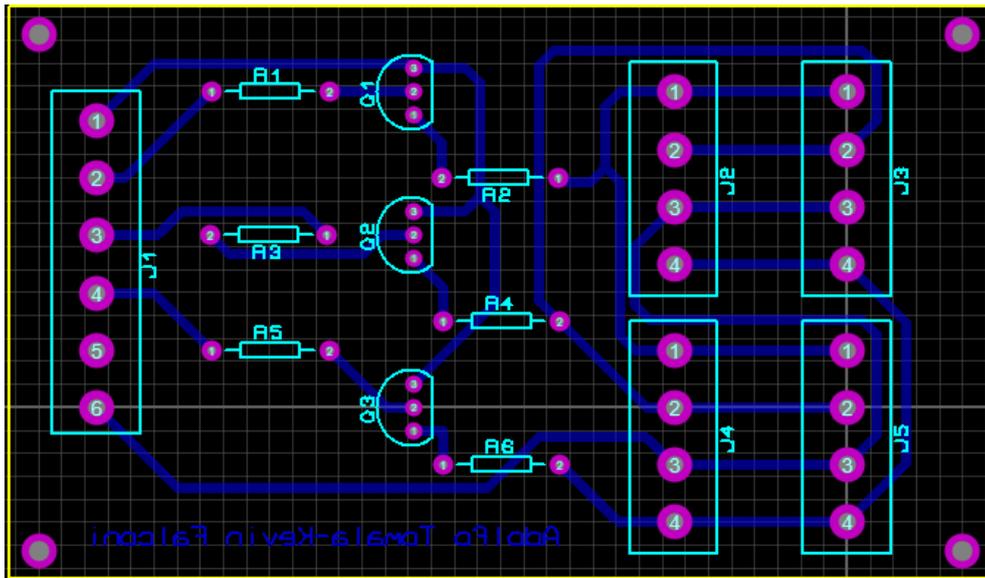


Figura 115. PCB de la placa de distribución de indicadores leds

DISEÑO DE PLACAS DE DISTRIBUCIÓN DE ACTUADORES Y SENSORES

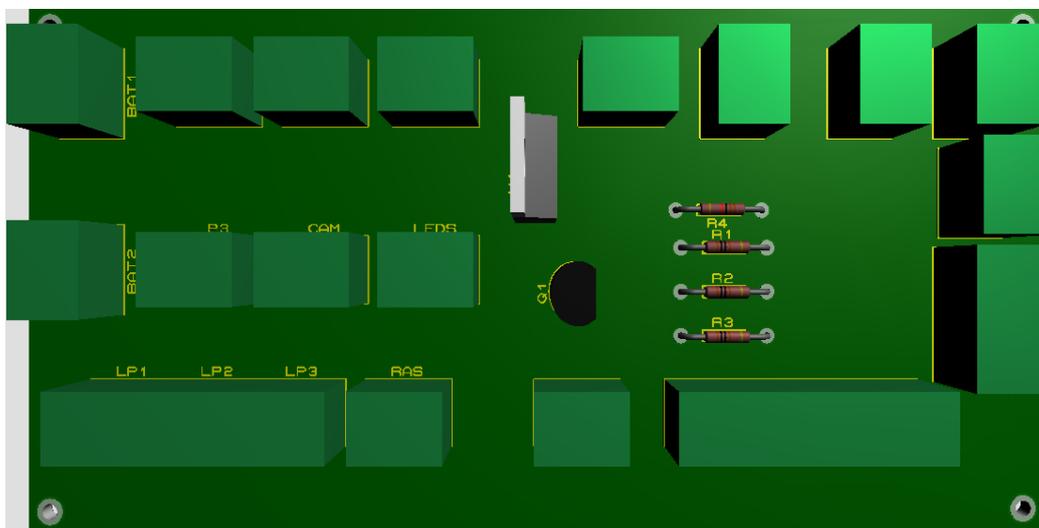


Figura 116. Vista superior de la placa de actuadores y sensores

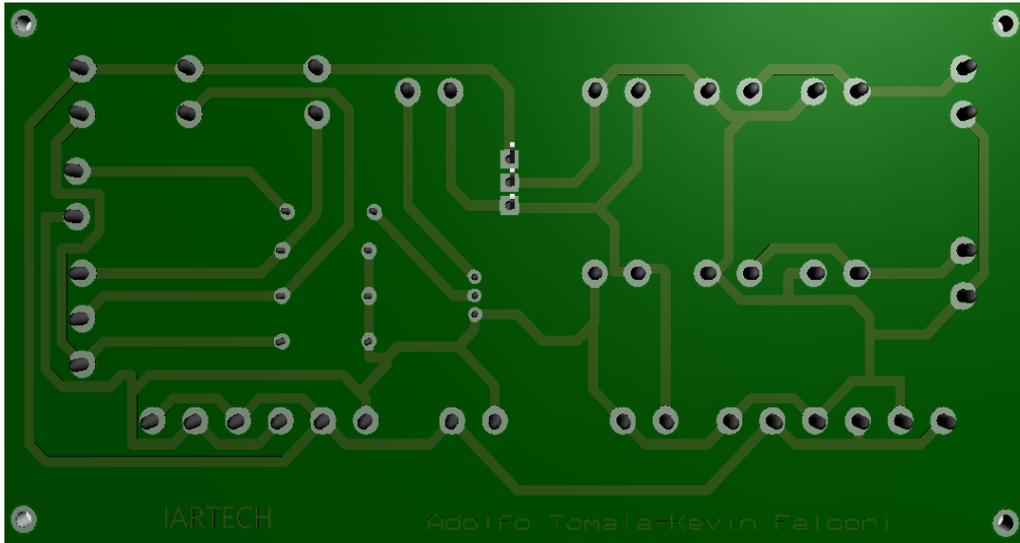


Figura 117. Vista inferior de la placa de actuadores y sensores

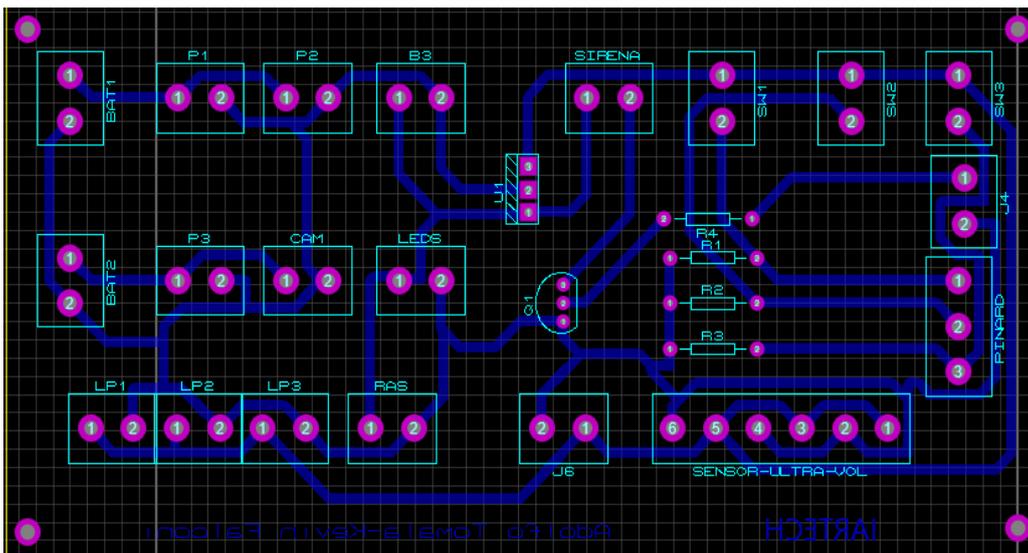


Figura 118. PCB de la placa de actuadores y sensores

COMPONENTES ELECTRÓNICOS:

Motor reductor de metal 19: 1 37Dx52L mm 12V (piñón helicoidal)

Especificaciones clave:

- Voltaje: 12 V
- Rendimiento sin carga: 530 RPM, 200 mA
- Extrapolación del estancamiento: 8.5 kg x cm (120 Oz x in). 5.5 A [29]

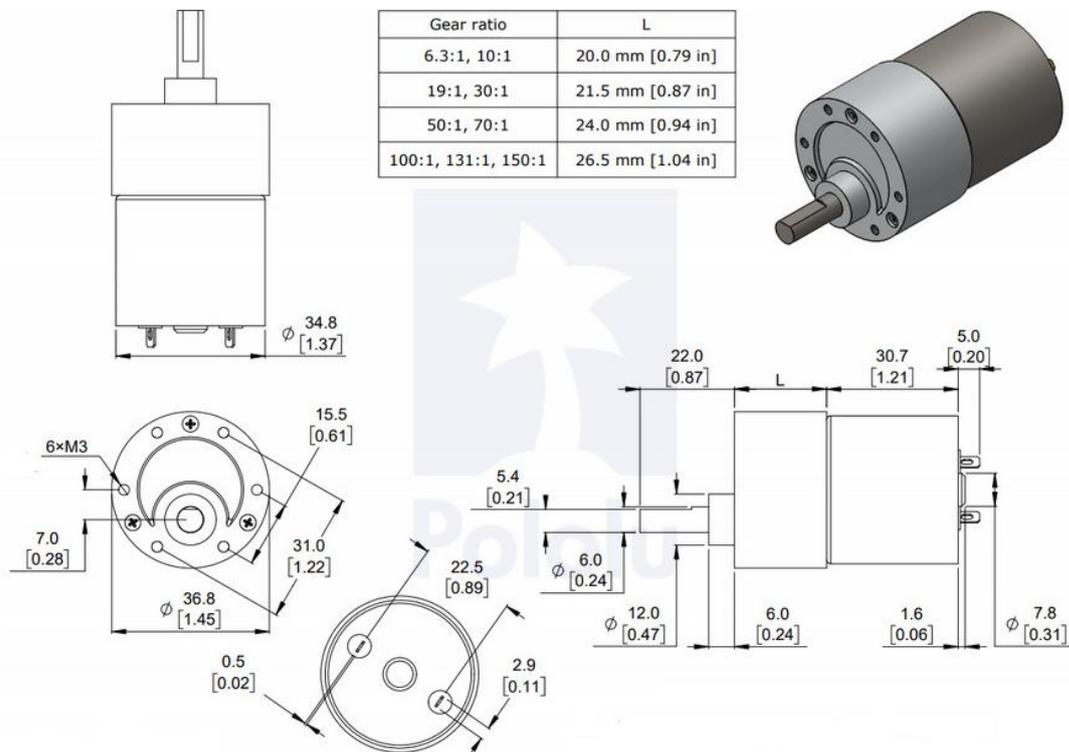


Figura 119. Dimensiones del motor reductor

8GB E14 Version Raspberry Pi 4 Model 4B BCM2711 Quad-core Cortex-A72
1.5GHz 8GB RAM with Dual Band WIFI Support POE



Especificaciones:

- Procesador: Broadcom BCM2711, Cortex-A72 de cuatro núcleos (ARM v8) SoC de 64 bits a 1,5 GHz
- Memoria RAM: 8GB LPDDR4
- Conectividad: LAN inalámbrica IEEE 802.11b / g / n / ac de 2.4 GHz y 5.0 GHz, Bluetooth 5.0, BLE
- 2 x USB 3.0
- 2 x USB 2.0
- GPIO: estándar de 40 pines
- Video y sonido: 2 puertos micro HDMI (hasta 4Kp60 admitidos)
- Puerto DSI para pantalla
- Puerto CSI para cámara
- Soporte de tarjeta SD: ranura para tarjeta micro SD para cargar el sistema operativo y almacenamiento de datos
- Potencia de entrada: 5V DC a través del conector USB-C (mínimo 3^a)
- 5V DC a través de los GPIO (mínimo 3^a)
- Power Over Ethernet (PoE) - habilitado (a través de un complemento PoE HAT por separado)
- Temperatura de funcionamiento 0–50°C.
- Dimensiones: (88 x 58 x 18,5mm)[30]

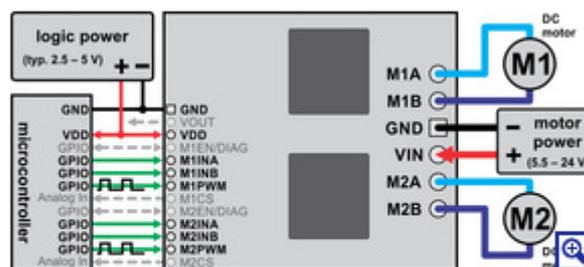
Pololu Dual VNH5019 Escudo del controlador del motor para Arduino



Especificaciones:

- Voltaje de funcionamiento: 5.5-24 V
- Resistencia a la on-resistencia del MOSFET (por pierna): 18 m Ω típico.
- Frecuencia pwm máxima: 20 kHz
- Sentido actual: 0.14 V/A típico
- Cierre por sobretensión: 24 V.min / 27 V tipo.
- Umbral alto de entrada lógica: 2.1 Vmin.
- Tiempo de sobrecalentamiento a 20 A: 20 s
- Tiempo de sobrecalentamiento a 15 A: años 90
- Corriente para tiempo de ejecución infinito: 12 A [31]

:



Doble escudo del controlador del motor
VNH5019 conectado a un
microcontrolador (las conexiones
grises son opcionales).

Figura 120. Conexión entre el arduino y driver de motor

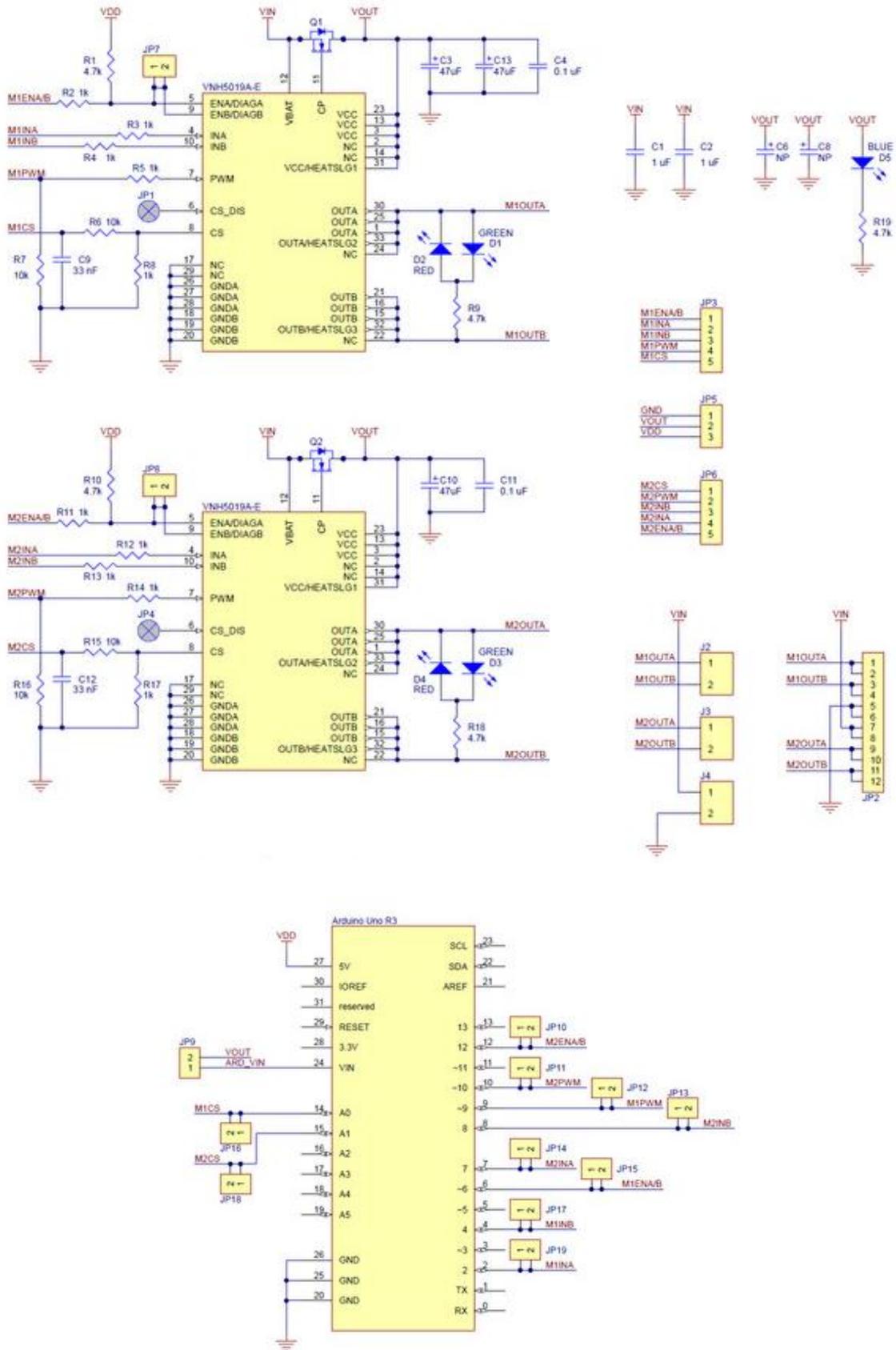


Figura 121. Diagrama esquemático del driver del motor

Bomba de agua dulce de 12 V CC con 2 abrazaderas de manguera Bomba de diafragma de 12 voltios Bomba de rociador autocebante con interruptor de presión 4.5 L/Min 1.2 GPM 80 PSI ajustable para RV Camper Marine Boat Lawn



Figura 122. Bomba de fumigación

Especificaciones:

- Altura de elevación máxima: 6.60 pies[32]
- Caudal máximo: 4.50 litros por minuto[32]
- Código UNSPSC: 40151500[32]
- Estilo: Bomba pulverizada de 12 Vcc[32]
- Material: Plástico [32]

Turnigy Graphene Panther 6000mAh 3S 75C Batería w / XT90



Figura 123. Fuente de alimentación (batería tipo grafeno)

Especificaciones:

- Capacidad: 6000 mAh [33]
- Recuento de celdas: 3 [33]
- Voltaje: 11.1 V [33]
- Descarga constante: 75 C [33]
- Descarga máxima (3 s): 150 C [33]
- Tamaño de la batería: 168x69x27 mm [33]
- Peso: 630 g [33]
- Resistencia del paquete: 7 M Ω [33]
- Enchufe de carga: JST-HX [33]
- Enchufe de descarga: XT90 [33]

Regulador de voltaje reducido del convertidor de CC LM2596S ajustable módulo de fuente de alimentación 36V 24V 12V a 5V 2ª Estabilizador de voltaje con pantalla digital voltímetro (6)

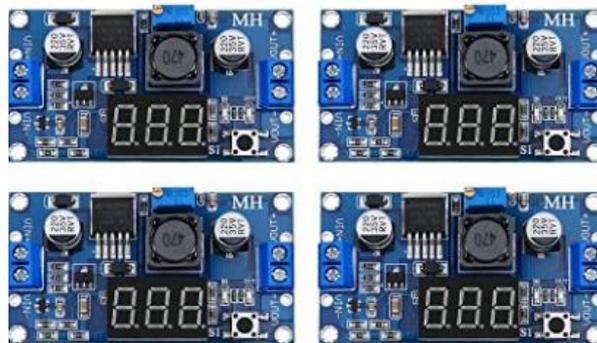


Figura 124. Regulador de voltaje

Modo de empleo:

- 1) Presione el botón en poco tiempo (1 segundo) para que IN led iluminado, el voltímetro muestra el valor de la tensión de entrada.
- 2) Presione el botón en un corto tiempo (1 segundo) de nuevo para que OUT led encendido, el voltímetro muestra el valor de la tensión de salida.

- 3) Gire el tornillo superior del potenciómetro en sentido horario para aumentar el valor de la tensión de salida, en sentido antihorario para reducirlo.
- 4) Presione el botón de entre 1 y 4 segundos para apagar el voltímetro. [34]

Función:

- Función de protección contra sobre temperatura incorporada
- Función de límite de corriente incorporada
- Función de protección contra cortocircuitos de salida incorporada
- Tamaño: aprox. 2.6 x 1.5 x 0.7 in. [34]

Sensor Horizontal de Nivel de Agua – Flotador Interruptor



Descripción:

- Se utiliza para detección del nivel de líquido de un tanque.[35]
- Puede activar una bomba, un indicador, una alarma u otro dispositivo.[35]
- Utilizable en aplicaciones con hidroponía, tanques de agua salada, tanques de agua dulce, jardinería, control de potencia para acuarios, etc.[35]
- Capacidad de los contactos: 70W.[35]
- Tensión de conmutación: 110V DC.[35]
- Corriente de conmutación: 0.5 A [35]

- Resistencia de aislamiento: $>10M\Omega$ [35]
- .Temperatura de funcionamiento: $-10 \sim 85\text{ }^{\circ}\text{C}$ [35]

Arduino Mega 2560



Arduino Mega posee las siguientes especificaciones:

- Microcontrolador: ATmega2560 [36]
- Voltaje Operativo: 5V [36]
- Voltaje de Entrada: 7-12V [36]
- Voltaje de Entrada(límites): 6-20V
- Pines digitales de Entrada/Salida: 54 (de los cuales 15 proveen salida PWM) [36]
- Pines análogos de entrada: 16 [36]
- Corriente DC por cada Pin Entrada/Salida: 40 mA [36]
- Corriente DC entregada en el Pin 3.3V: 50 mA [36]
- Memoria Flash: 256 KB (8KB usados por el bootloader) [36]
- SRAM: 8KB [36]
- EEPROM: 4KB [36]
- Clock Speed: 16 MHz [36]

Sensor ultrasónico HC-SR04



Características técnicas del sensor ultrasonido HC-SR04

- Voltaje operativo de 5 VDC [37].
- Intensidad de menos de 2 mA. [37]
- Ángulo de detección de no más de 15° [37]
- Distancia de detección de 2 cm hasta 450 cm. [37]
- Precisión de hasta 3 mm. [37]
- Cables de conexión Macho-Hembra de 220 mm. [37]