



INSTITUTO SUPERIOR TECNOLÓGICO
VIDA NUEVA

CARRERA:

TECNOLOGÍA EN ELECTROMECAÁNICA

DISEÑO Y CONSTRUCCIÓN DE UN ROBOT DE DESINFECCIÓN DE
SUPERFICIES EN AMBIENTES CERRADOS POR MEDIO DE RAYOS
ULTRAVIOLETAS CONTROLADO MEDIANTE UNA APP.

“TRABAJO INTEGRADOR DE GRADO”

PRESENTADO POR:

JAMI QUINATO A CRISTIAN JAVIER

TUTOR

ING. TOCA TRUJILLO LUIS EDISON

FECHA:

FEBRERO, 2021

Ing. Luis Toca
0984006074
1719889253
Aprobado

QUITO - ECUADOR

CESIÓN DE DERECHOS DE AUTOR

Yo, JAMI QUINATO A CRISTIAN JAVIER portador/a de la cédula de ciudadanía 0504151739, facultado/a de la carrera TECNOLOGÍA EN ELECTROMECAÁNICA, autor/a de esta obra certifico y proveo al Instituto Superior Tecnológico Vida Nueva, usar plenamente el contenido del informe con el tema DISEÑO Y CONSTRUCCIÓN DE UN ROBOT DE DESINFECCIÓN DE SUPERFICIES EN AMBIENTES CERRADOS POR MEDIO DE RAYOS ULTRAVIOLETAS CONTROLADO MEDIANTE UNA APP, con el objeto de aportar y promover la lectura e investigación, autorizando la publicación de mi proyecto de titulación en la colección digital del repositorio institucional bajo la licencia de Creative Commons: Atribución-No Comercial-Sin Derivadas.

En la ciudad de Quito, del mes de Febrero del 2021.

JAMI QUINATO A CRISTIAN JAVIER

C.C.0504151739

CERTIFICACIÓN

En mi calidad de Tutor del Proyecto: DISEÑO Y CONSTRUCCIÓN DE UN ROBOT DE DESINFECCIÓN DE SUPERFICIES EN AMBIENTES CERRADOS POR MEDIO DE RAYOS ULTRAVIOLETAS CONTROLADO MEDIANTE UNA APP en la ciudad de Quito, presentado por el ciudadano JAMI QUINATO A CRISTIAN JAVIER, para optar por el título de Tecnólogo en Electromecánica, certifico, que dicho proyecto ha sido revisado en todas sus partes y considero que reúne los requisitos y méritos suficientes para ser sometido a la presentación pública y evaluación por parte del tribunal examinador que se designe

En la ciudad de Quito, del mes de Febrero del 2021.

TUTOR:

ING. TOCA TUJILLO LUIS EDISON

APROBACIÓN DEL JURADO EXAMINADOR O TRIBUNAL

Los miembros del jurado examinador aprueban el informe de investigación, sobre el tema: DISEÑO Y CONSTRUCCIÓN DE UN ROBOT DE DESINFECCIÓN DE SUPERFICIES EN AMBIENTES CERRADOS POR MEDIO DE RAYOS ULTRAVIOLETAS CONTROLADO MEDIANTE UNA APP en la ciudad de Quito Del estudiante: Jami Quinatoa Cristian Javier de la Carrera en Tecnología en Electromecánica

Para constancia firman:

.....
CI:

.....
CI:

.....
CI:

.....
CI:

DECLARACIÓN DE AUTENTICIDAD

Yo, Jami Quinatoa Cristian Javier con cédula de ciudadanía 0504151739, estudiante del Instituto Superior Tecnológico Vida Nueva, declaro que he realizado este trabajo de titulación tomando en consideración citas bibliográficas que se nombran en este texto.

El Instituto Tecnológico Superior Vida Nueva puede utilizar este trabajo de titulación como una ayuda bibliográfica.

JAMI QUINATO A CRISTIAN JAVIER

DEDICATORIA

“A cada acción tiene su reacción”

AGRADECIMIENTOS

A mi familia que a pesar de todo me han apoyado por seguir adelante y son el motivo de seguir creciendo profesionalmente y en especial a mi padre y madre por sus consejos de vida y la lucha constante de seguir en pie.

A mis maestros que conforman la carrera de Electromecánica que han impartido todo sobre la carrera y sus anécdotas referente a ello para poder ser un profesional competitivo. Y a mi tutor Ing. Luis Toca que me permitió conocer sobre la robótica.

Al ISTVN por permitirme estudiar utilizando las aulas y talleres que cuenta la institución y adquirir todos los conocimientos necesarios en la carrera de Tecnología en Electromecánica.

ÍNDICES

Índice de contenido

CESIÓN DE DERECHOS DE AUTOR.....	I
CERTIFICACIÓN	II
APROBACIÓN DEL JURADO EXAMINADOR O TRIBUNAL	III
DECLARACIÓN DE AUTENTICIDAD	IV
DEDICATORIA	V
AGRADECIMIENTOS	VI
RESUMEN.....	XI
ABSTRACT.....	XII
INTRODUCCIÓN	XIII
CAPÍTULO I.....	1
MARCO TEÓRICO.....	1
COVID – 19	1
Robots en la medicina	1
Da Vinci	2
Robot desinfectador	3
Definición de términos básicos	4
Luz ultravioleta	4
NodeMCU ESP 8266	6
Motorreductor	7
Driver Monster	8
Inversor	9
Lámpara ultravioleta	10
Batería	11
CAPÍTULO II	12
METODOLOGÍA DEL PROYECTO INTEGRADOR-LA INVESTIGACIÓN	12
Diseño metodológico	12
Variables y definición operacional.....	12
Técnicas de recolección de datos	12
CAPÍTULO III.....	14
PROPUESTA.....	14
Diseño eléctrico – electrónico	14
Esquemático electrónico	15
Diseño de la PCB	16
Diseño Mecánico.....	19
Diseño de la base de los elementos	19
Estructura Externa	20
Ensamblaje eléctrico-electrónico	21

Ensamblaje Mecánico	22
Programación de la placa de desarrollo.....	23
Importación de librerías	23
Declaración de variables	23
Creación de la Función Void Setup.....	24
Conexión Wifi.....	24
Creación de funciones	25
Creación de la función Void gosw	26
Creación de la función Void loop	26
Desarrollo de la App	27
Diagrama de bloques.....	28
CAPÍTULO IV	29
RESULTADOS	29
CONCLUSIONES	30
RECOMENDACIONES	30
ANEXOS.....	31
Anexo 1: Código de programación del robot	31
Anexo 2: Planos de la estructura	35
Anexo 3: PCB del circuito electrónico.....	36
Anexo 4: PCB en 3D.....	36
Bibliografía	37

Índice de figuras

Figura 1. Robot Tug	2
Figura 2. Robot Da Vinci	2
Figura 3. Exoesqueletos robóticos	3
Figura 4. Robot XDBOT	3
Figura 5. UVR-bot	4
Figura 6. Luz ultravioleta	4
Figura 7. NodeMCU ESP8266	6
Figura 8. Motorreductor Dc	7
Figura 9. Driver Monster	8
Figura 10. Inversor 500W	9
Figura 11. Lámpara ultravioleta	10
Figura 12. Batería plomo ácido	11
Figura 13. Diagrama de bloques del proceso de funcionamiento	14
Figura 14. Diagrama de bloques del sistema de carga	15
Figura 15. Diseño esquemático en Isis Proteus	15
Figura 16. Diseño esquemático en Ares Proteus	16
Figura 17. Visualización en 3D Proteus	17
Figura 18. Importación del diseño en PDF	17
Figura 19. Importación de serigrafía en PDF	18
Figura 20. Diseño de la base 3D en SolidWorks	19
Figura 21. Diseño de la estructura 3D en SolidWorks	20
Figura 22. Ensamblaje eléctrico	21
Figura 23. Ensamblaje eléctrico	21
Figura 24. Ensamblaje del robot	22
Figura 25. Ensamblaje final del robot	22
Figura 26. Declaración de Librerías en Arduino	23
Figura 27. Creación de variables en Arduino	23
Figura 28. Creación de función void setup en Arduino	24
Figura 29. Conexión wifia en Arduino	24
Figura 30. Creación de funciones Void en Arduino	25
Figura 31. Creación de función Void gosw Arduino	26
Figura 32. Creación de función Void Loop Arduino	26
Figura 33. Pantalla principal de la App	27
Figura 34. Desarrollo de la App	27
Figura 35. Diagrama de bloques	28
Figura 36. Ícono de la aplicación	28

Índice de tablas

Tabla 1 <i>Características del NodeMCU ESP8266</i>	6
Tabla 2 <i>Características del Motorreductor 30:1</i>	7
Tabla 3 <i>Características del Driver Monster</i>	8
Tabla 4 <i>Características del Inversor</i>	9
Tabla 5 Características de la lámpara ultravioleta.....	10
Tabla 6 <i>Características de la batería</i>	11

RESUMEN

El objetivo central del robot de desinfección fue por la mejora en la eficiencia de la desinfección a medida que crece la tecnología, por ello la necesidad de una extensa investigación sobre los métodos de desinfección. Desde hace varios años la luz ultravioleta ha sido ampliamente usada para esterilizar pequeños flujos residenciales, así como también grandes flujos en proyectos comerciales e industriales; pero hoy es un método efectivo para prevenir la propagación de COVID-19. Por tal motivo el presente proyecto está basado en el mismo método para la construcción del robot de desinfección, en este sentido el proyecto es capaz de propiciar en la búsqueda de información a través de un método de investigación y registro de diferentes prototipos usados para la desinfección en el área de la salud.

Con estos antecedentes se procede a diseñar la base que complemente la estructura y sirva para acoplar los distintos elementos eléctricos, electrónicos y la verificación de estabilidad para el armado del robot, se procede a la programación de la placa de desarrollo y su respectiva aplicación que permite el control a distancia del robot.

Palabras claves: Robot desinfectante, COVID -19, rayo ultravioleta, métodos de desinfección.

ABSTRACT

The main objective of the disinfection robot was to improve disinfection efficiency as technology grows, hence the need for extensive research on disinfection methods. For several years now, ultraviolet light has been widely used to sterilize small residential flows, as well as large flows in commercial and industrial projects. But today, it is an effective method to prevent the spread of COVID-19. For this reason, the current project is based on the same method for the construction of the disinfection robot. In this sense, the project is able to provide an information search through a research and registration method of different prototypes used for disinfection in the health area.

Against this background, it proceeds to design the base that complements the structure and serves to attach the different electrical and electronic elements and the stability verification for the robot assembly. It proceeds to the programming of the development board and its respective application that allows the robot to be remotely controlled.

Keywords: Disinfectant robot, COVID -19, ultraviolet ray, disinfection methods.



Lcdo. Ricardo Quishpe

INTRODUCCIÓN

La robótica es una de las industrias que más se está desarrollando en los últimos años y ahora, con la crisis provocada por el Covid-19, los robots son los encomendados como una solución para evitar el contacto físico e incluso para desinfectar las zonas públicas.

Los primeros en abrazar esta tecnología son las empresas chinas que han desarrollado todo tipo de objetos sin la necesidad de un conductor ni de contacto humano, totalmente autónomos, desarrolladas para cada área en específico que son de baja media y alta velocidad dependiendo el trabajo a realizarse. Pero lo más interesante es que incorporan un sistema de desinfección para los espacios públicos.

Al igual que la mecánica, la electrónica y la programación han hecho su trabajo por un lado las investigaciones giran en torno a tener componentes electrónicos e incluso componentes electromecánicos capaces de aguantar la exposición a diferentes ambientes. Con la ayuda de diferentes algoritmos de programación ha sido posible calibrar velocidades, censar distancias, generar determinados ángulos de giro para las extremidades de los diferentes prototipos.

En esta dirección es posible entender la presente investigación como aporte a las investigaciones sobre los robots desinfectadores. Estos robots han sido poco estudiados a nivel local, sin embargo, su uso tiene preferencias, quizás uno de los principales problemas es la dificultad de encontrar una acogida adecuada donde sea demandada este tipo de robots actualmente su acogida es en los hospitales, sin ser conocido en otras áreas que no sea de la salud.

Metodológicamente este proyecto fue realizado a base de una investigación experimental la cual consta de una amplia búsqueda de las principales tecnologías usadas para estos robots desinfectadores, así como tener ciencia de los antecedentes o investigaciones realizadas en el país.

En el país se conoce sobre el uso de la robótica para la desinfección, mediante investigaciones se pudo conocer que el uso de rayos ultravioletas ya se usaba desde años anteriores para la desinfección en hospitales, en zonas donde la higiene es primordial, pero el conocimiento de este método se ha quedado únicamente en los hospitales sin ser

dado a conocer en otros sectores.

De este modo se puede describir este trabajo en cuatro capítulos, el primero está englobado en la búsqueda sobre los robots en la medicina y desinfección, las aplicaciones específicas que han tenido estos robots al igual de un glosario de términos, este último recurso permite comprender de una mejor manera todos los conceptos teóricos y definiciones de varios elementos eléctricos, electrónicos y mecánicos que comprende el robot.

En el segundo capítulo se presenta la metodología usada en la investigación y las diferentes técnicas que se pueden aplicar en este tipo de proyectos. Para ello la necesidad de aplicar la observación a fin de visualizar el desarrollo que se presente en el robot y el desempeño en los laboratorios de pruebas con el fin de recabar información para su posterior mejora con la ayuda de búsquedas en portales confiables.

En el tercer capítulo considerado como el capítulo en la cual se realiza la construcción del robot, existen subcapítulos importantes donde se presenta todos los diseños, eléctricos, electrónico y mecánico. Todos los diseños fueron creados en programas CAD donde se puede apreciar el acabado de cada uno de los diseños realizados para posteriormente ser materializados con todos los componentes necesarios. Se realiza una simulación previa al funcionamiento, este puede ser mediante software o físico, una vez comprobado su funcionamiento se procede al ensamblaje, empezando por lo mecánico, construyendo la base y la estructura utilizando fijaciones para mantener unida toda la estructura. En el ensamblaje eléctrico se ubica todos los componentes y se realiza las respectivas conexiones utilizando terminales adecuados para unir todo el sistema eléctrico, también el uso de sujetadores para los cables que sea ordenado. Para el sistema electrónico se realiza la PCB mediante el método de planchado y se suelda cada uno de los componentes electrónicos seguido de la comprobación del funcionamiento de la placa completa.

Una vez ensamblado todos los componentes se realiza la respectiva prueba de funcionamiento del robot y denota las fallas para ser corregidas.

El último capítulo comprende a los resultados, conclusiones y recomendaciones que genera este prototipo, entre los resultados obtenidos se puede precisar el indispensable trabajo que deben tener las estructuras mecánicas y equilibrio del robot en movimiento.

CAPÍTULO I

MARCO TEÓRICO

COVID – 19

Los coronavirus son una extensa familia de virus que pueden causar enfermedades tanto en animales como en humanos. En los humanos, se sabe que varios coronavirus causan infecciones respiratorias. El coronavirus que se ha descubierto más recientemente causa la enfermedad por coronavirus COVID-19. (OMS, 2019)

El COVID-19 es la enfermedad infecciosa causada por el coronavirus que se ha descubierto más recientemente. Tanto este nuevo virus como la enfermedad que provoca, eran desconocidos antes de que estallara el brote en Wuhan (China) en diciembre de 2019. Actualmente el COVID-19 es una pandemia que afecta a muchos países de todo el mundo.

Robots en la medicina

Los robots continúan avanzando a pasos agigantados en su desarrollo y la medicina es uno de los sectores donde más rendimiento producen. En el pasado siglo, los robots empezaron a realizar muchas labores en el campo industrial, pero para este siglo XXI van incorporándose a muchas más tareas. La medicina robótica cuenta con ejemplos de robots que ya son una ayuda imprescindible para los médicos. (Robot esfera, 2018)

Tug

Además de curar, se usan robots para hacer más liviana la carga y algunos procesos para el paciente. Tug es ejemplo de ello, pues lleva medicamentos al personal del hospital, lo que hace que lleguen más rápido y evitando los errores en las entregas de los mismos.



Figura 1. Robot Tug

Fuente: Robotesfera

Da Vinci

Este robot médico nacido en el año 2000 fue el primer robot cirujano. Actualmente se usa en todo el mundo a la hora de realizar cirugías complejas de manera menos invasiva. La gran capacidad de este robot en cuanto a precisión hace que llegue a niveles complicados de lograr por un ser humano, reduciendo pérdidas de sangre, dolores postoperatorios y demás riesgos en relación con las anestesias.



Figura 2. Robot Da Vinci

Fuente: Robotesfera

Exoesqueletos robóticos

Muchas personas pensaban que nunca caminaría, actualmente andan con exoesqueletos robóticos, que rodean el tronco y extremidades del cuerpo, imitando los movimientos del cuerpo humano que son ordenados desde el cerebro de la persona.



Figura 3. Exoesqueletos robóticos

Fuente: Robotesfera

Robot desinfectador

eXtreme Disinfection roBOT (XDBOT)

El robot se puede operar de forma inalámbrica con una computadora portátil o tableta. Con su brazo robótico de 6 ejes, el XDBOT puede imitar el movimiento humano para llegar a lugares incómodos y una cámara de alta definición, el robot puede navegar de forma semiautónoma en cualquier entorno. (SAKHARKAR, 2020)



Figura 4. Robot XDBOT

Fuente: TECH EXPLORIST

UVR-bot

Mediante un escáner de láser 3D, el UVR-bot analiza toda la habitación, calcula el tiempo y la posición donde tiene que estar, y emite radiación UV-C para eliminar el ADN y ARN de virus y bacterias. La desinfección del espacio puede tardar entre 5 y 15 minutos, dependiendo del tamaño del lugar. (Clarín, 2020)



Figura 5. UVR-bot

Fuente: Clarín

Definición de términos básicos

Luz ultravioleta

Una longitud de onda específica de luz ultravioleta, que ahora se puede implementar a través de fibras óptica de difusión de luz, es altamente eficiente para matar bacterias resistentes a los medicamentos. La tecnología está diseñada para prevenir infecciones alrededor de los dispositivos o instrumentales médicos que penetran en la piel, como las catéteres y cánulas de dispositivos cardiacos. (Álvarez, 2018)

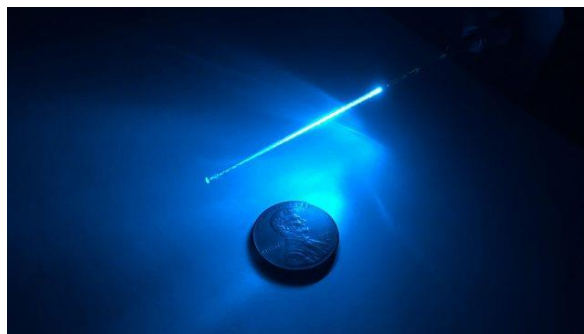


Figura 6. Luz ultravioleta

Fuente: smartlighting

Un estudio realizado por investigadores de la Universidad de Hiroshima descubrió que el uso de luz ultravioleta C con una longitud de onda de 222 nanómetros mata al SARS-CoV-2, sin dañar las células vivas del ojo y la piel en seres humanos.

De acuerdo al Departamento de Relaciones Públicas de la Universidad de Hiroshima, es la primera investigación en el mundo que demuestra la eficacia de esta tecnología contra el virus que causa la COVID-19. El experimento in vitro mostró que el 99.7% del cultivo viral del SARS-CoV-2 murió después de una exposición de 30 segundos a una irradiación UV-C de 222 nanómetros a 0.1 mW / cm².

222 nm frente a 254 nm UVC

Una longitud de onda de 222 nanómetros de UV-C no puede penetrar la capa superior del ojo y la piel humanos, por lo que no dañará las células vivas que se encuentran debajo. Esto la convierte en una alternativa más segura pero igualmente potente a las lámparas germicidas UVC de 254 nanómetros, más dañinas, que se utilizan cada vez más en la desinfección de las instalaciones sanitarias. (Serrano, 2020)

NodeMCU ESP 8266

Es una pequeña placa Wifi lista para usar en cualquier proyecto IoT. Está montada alrededor del ya conocido ESP8266 y expone todos sus pines en los laterales. Además, ofrece más ventajas como la incorporación de un regulador de voltaje integrado, así como un puerto USB de programación. Se puede programar con LUA o mediante el IDE de Arduino. (Llamas, 2018)



Figura 7. NodeMCU ESP8266

Fuente: Elaboración propia

Tabla 1

Características del NodeMCU ESP8266

CPU	Tensilica Xtensa LX3 (32 bit)
Placa	NodeMCU v2 (Amica)
Voltaje de operación	5-10V DC
Voltaje de Entradas/Salidas	3.3V DC (No usar 5V)
Frecuencia de reloj	80MHz/160MHz
Data RAM	96KB
Memoria Flash Externa	4MB
Pines Digitales GPIO	17 (4 pueden configurarse como PWM a 3.3V)
Pin Analógico ADC	1 (0-1V)
DC corriente pines	12mA
Wifi	Direct (P2P), soft-AP
Módulo wifi	2.4 GHz
Leds indicadores	2
Dimensiones	49*26*12 mm
Peso	9 gramos

Fuente. Elaboración propia

Motorreductor

Este motorreductor es un potente motor DC cepillado con 30: 1 caja de engranajes de metal destinado para funcionar a 12V. La caja de cambios está compuesta principalmente de engranajes rectos, pero cuenta con engranajes helicoidales para la primera etapa para reducir el ruido y mejorar la eficiencia. (Pololu, 2019)



Figura 8. Motorreductor Dc

Fuente: Elaboración propia

Tabla 2

Características del *Motorreductor 30:1*

Nombre Técnico	Motorreductor
Marca	Pololu
Tamaño	37 x 52 x 16 mm
Peso	185 g
Diámetro de eje	6 mm
Relación de transmisión	30:1
Velocidad sin carga @ 12V	330 RPM
Corriente sin carga @ 12V	0.2 A
Corriente con carga Máx @ 12V	5.5 A
Torque @ 12V	190 oz·in

Fuente. Elaboración propia

Driver Monster

Es un controlador de motor de puente completo diseñado para una amplia gama de aplicaciones automotrices. El dispositivo incorpora un controlador de lado alto monolítico dual y dos interruptores de lado bajo. (Mybotic, 2018)

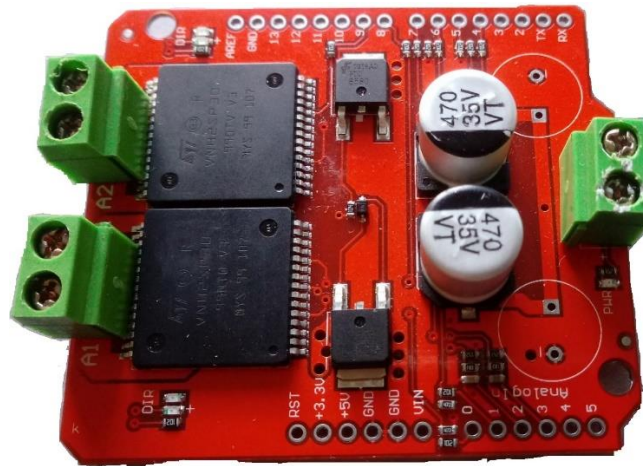


Figura 9. Driver Monster

Fuente: Elaboración propia

Tabla 3

Características del *Driver Monster*

Nombre	Driver Monster Motor Shield VNH2SP30
Circuito integrado principal	VNH2SP30
Corriente pico de operación	30 A (incluir disipador a partir de 14 A)
Corriente constante de operación	14 A
Rango de Voltaje	5.5 V – 16 V
Sensado de corriente en pin análogo	
MOSFET en resistencia	19 mΩ
Apagado térmico	
Apagado por subtensión y sobretensión	

Fuente. Elaboración propia

Inversor

Un inversor de corriente es un aparato electrónico. La función del inversor es cambiar un voltaje de entrada de corriente continua a un voltaje simétrico de salida de corriente alterna, con la magnitud y frecuencia deseada por el usuario. Los inversores de corriente también reciben el nombre de convertidores de corriente y onduladores de corriente. (Planas, 2016)



Figura 10. Inversor 500W

Fuente: Elaboración propia

Tabla 4

Características del Inversor

Marca	BELTTT
Tensión de entrada	12 V DC
Tensión de salida	110 V AC
Frecuencia de salida	50Hz/60Hz
Potencia	500 W
Tipo de onda	Onda Modificada
Tipo de batería	Batería de ácido sólido
Dimensiones	152 x 89 x 47 mm
Peso	408 g
Función de protección	Cortocircuito \ sobrecarga \ sobre tensión

Fuente. Elaboración propia

Lámpara ultravioleta

Las lámparas ultravioletas de media presión constan de un bulbo o tubo de cristal de cuarzo relleno de gas, con dos electrodos en los extremos, que al suministrarle electricidad forman un arco eléctrico entre ellos, calentando y subiendo la presión de dicho gas y produciendo la emisión de luz. En función de los gases y aditivos que contenga la lámpara UV se obtendrá un espectro u otro. (BCB, 2019)

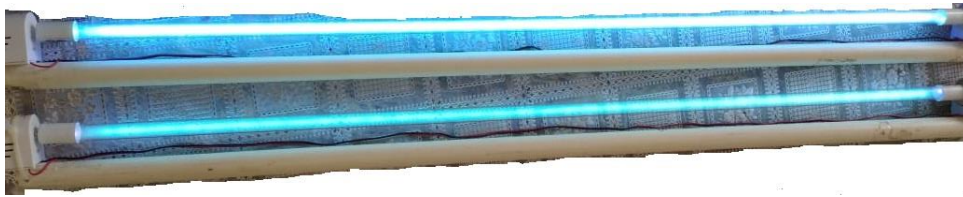


Figura 11. Lámpara ultravioleta

Fuente: Elaboración propia

Tabla 5

Características de la lámpara ultravioleta

Marca	JOSTIN
Tensión	110 V AC
Potencia	40W
Frecuencia	50Hz/60Hz
Longitud de onda	254 nm
Tipo de gas	Gas de mercurio
Aplicaciones	Viviendas, Restaurantes y comercio en general
Dimensiones	119 mm x 20 mm
Peso	158 g
Incluye balastro	

Fuente. Elaboración propia

Batería

Batería de plomo ácido, que el rendimiento está especialmente diseñado es adecuado para la aplicación de generación fotovoltaica. La carga de la batería y el control de la descarga con la potencia máxima del generador fotovoltaico es el punto clave para aumentar la eficiencia del sistema de generación. (Li, 2016)



Figura 12. Batería plomo ácido

Fuente: Elaboración propia

Tabla 6

Características de la batería

Marca	First Power
Modelo	FP1290
Voltaje	12 V
Capacidad	9 Ah
Tipo	Batería de Plomo-Ácido
Recargable	Si
Standby use	13.6 – 13.8 V
Max initial Current:	Les tan 2.7 A
Dimensiones	150 x 65 x 94 mm
Peso	2.5 kg

Fuente. Elaboración propia

CAPÍTULO II

METODOLOGÍA DEL PROYECTO INTEGRADOR-LA INVESTIGACIÓN

Diseño metodológico

Según lo establecido para la construcción de este proyecto, se utilizará una investigación experimental basada en recolección de información búsqueda de fuentes teóricas, se procederá a realizar el respectivo presupuesto y verificar alcances técnicos y aportes académicos para futuros proyectos de mismo tipo. Se procederá a recolectar información para el diseño del robot que permita una óptima estabilidad y movilidad, para posteriormente diseñar el sistema eléctrico – electrónico seguido de la programación de la placa de desarrollo y el diseño de la App para el respectivo control del robot.

Variables y definición operacional

Las variables que se presentan en este proyecto son de carácter empírico y experimental. En las que se encuentran dos variables la más importante es el peso para poder elegir los motores adecuados y la fuente de energía eléctrica que consumirá el robot. Otra de las variables es la estabilidad para que el robot mantenga una estabilidad con respecto a la superficie.

Técnicas de recolección de datos

En este proyecto se utilizarán las técnicas de observación y registros. La técnica de observación se basa en el conocimiento empírico, es decir, en las experiencias de proyectos anteriores desarrollados durante el periodo académico y el fortalecimiento de conocimientos en los concursos de robótica con resultados favorables que se obtuvieron, se pretende llevar a siguiente nivel este tipo de proyecto.

La técnica de registro permite generar búsquedas de estudios y proyectos empresariales realizados durante los últimos diez años con relación a los robots de desinfección utilizando rayos ultravioletas.

CAPÍTULO III

PROPUESTA

Diseño eléctrico – electrónico

Dentro de este apartado se presentan todos los diagramas eléctricos y electrónicos de los elementos. Para el diseño eléctrico – electrónico se presenta un diagrama de bloques de cómo es el proceso de funcionamiento del robot desinfectador.

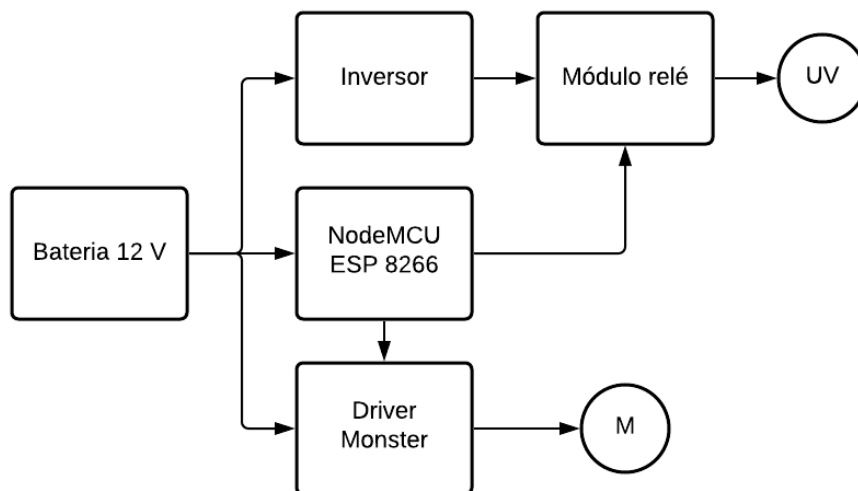


Figura 13. Diagrama de bloques del proceso de funcionamiento

Fuente: Elaboración propia

El robot tiene implementado un sistema de carga para sus baterías, que a su vez puede estar cargando las baterías mientras se desinfecta en un área reducida donde el robot tiene que permanecer estático en un solo punto. Para el sistema de carga, debe seleccionar el mecanismo de control en el modo que se desea activar el robot.

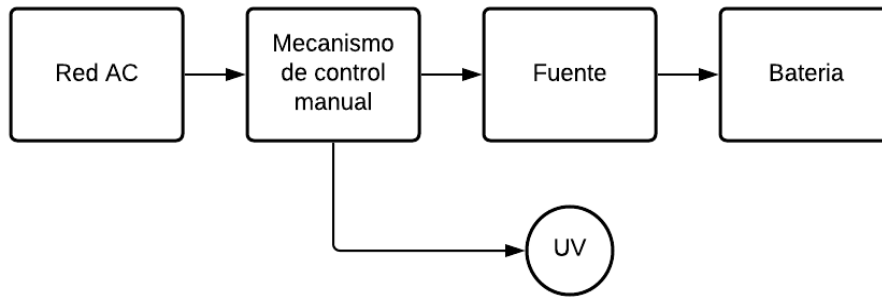


Figura 14. Diagrama de bloques del sistema de carga

Fuente: Elaboración propia

Esquemático electrónico

Para el esquemático electrónico se utilizó el software de Proteus, seleccionando todos los componentes necesarios para la placa. Se procede a unir todos los componentes utilizando marcas para evitar los diferentes trazos con el fin de que el diseño esquemático tenga un buen aspecto, desde la alimentación hasta el control de la carga que son las lámparas ultravioletas y obtener el esquemático electrónico de la placa electrónica del robot.

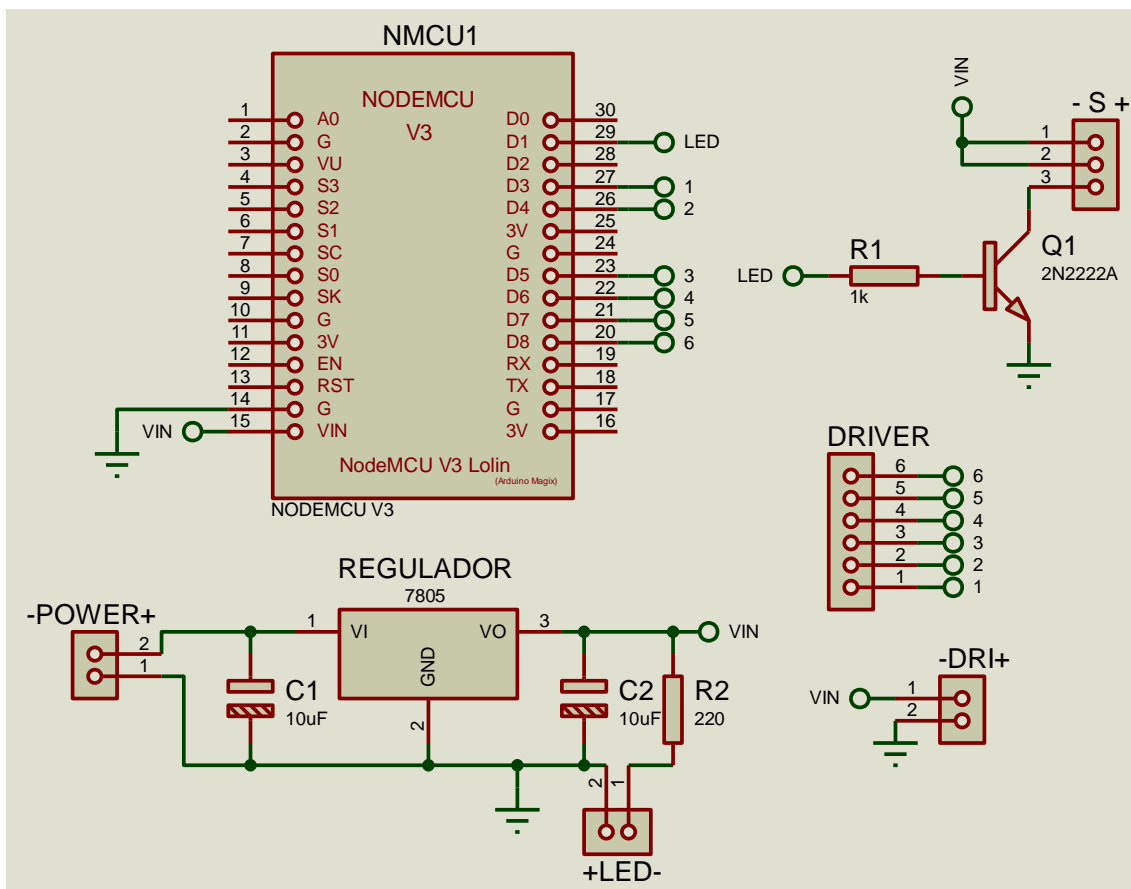


Figura 15. Diseño esquemático en Isis Proteus

Fuente: Elaboración propia

Se instaló la librería del NodeMCU para realizar el diagrama, se realiza una etapa de regulador de voltaje por motivo que la placa NodeMCU admite un voltaje en un rango de 5 – 10 V DC y la alimentación para los demás componentes son de 5 V DC y se utilizó un regulador LM7508 a 1A.

Diseño de la PCB

Para el diseño de la placa se lo realiza en el mismo software una vez terminado el esquemático en Isis se procede a pasar los componentes en Ares. Realizando un área donde se añaden los componentes, con las herramientas de Proteus se realiza la ubicación correspondiente de los componentes, los trazados y configuración de las pistas. Por último, se inserta texto y un logo a la placa y se importa en formato PDF para su posterior impresión. La placa se realiza por el método de planchado.

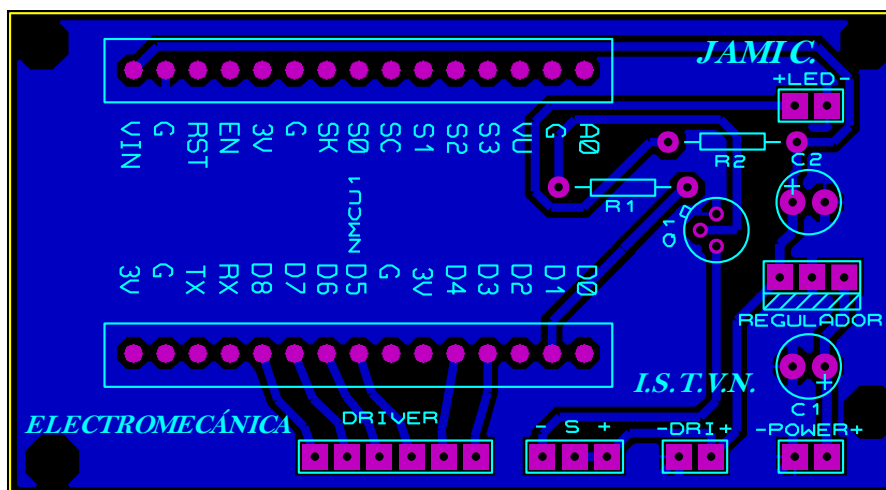


Figura 16. Diseño esquemático en Ares Proteus

Fuente: Elaboración propia

Una vez terminado de realizar todas las configuraciones de la placa y estar conforme con lo realizado, Proteus al igual que el demás software tiene la opción de apreciar el modelo la placa en 3D como se muestra a continuación.

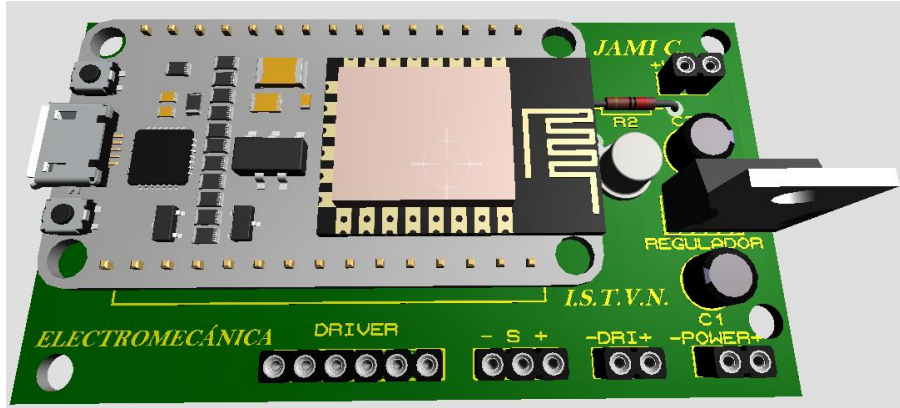


Figura 17. Visualización en 3D Proteus

Fuente: Elaboración propia

Para la generación de los archivos en PDF y pasarlo a la PCB se genera el primer archivo que es de la parte inferior de la PCB, que es la capa de cobre de la placa. Una vez generado se pasa el diseño a la placa con el método de planchado que ha sido mencionado anteriormente. Hay que destacar que los acabados de la placa son muy semejantes a las placas profesionales.

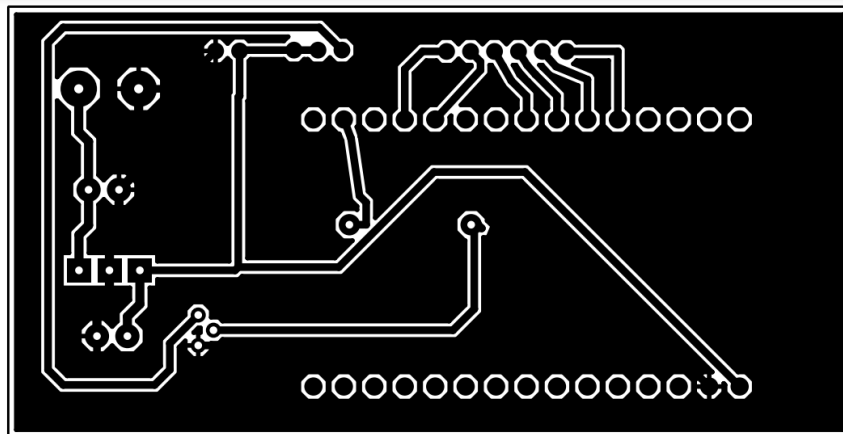


Figura 18. Importación del diseño en PDF

Fuente: Elaboración propia

El segundo archivo trata sobre la serigrafía que son los textos para la parte superior de la PCB para poder identificar el lugar donde van insertados los componentes al momento de soldar a la PCB se lo realiza con las configuraciones correspondientes. Al momento de generar el archivo se lo realiza en la configuración de espejo con el texto invertido.

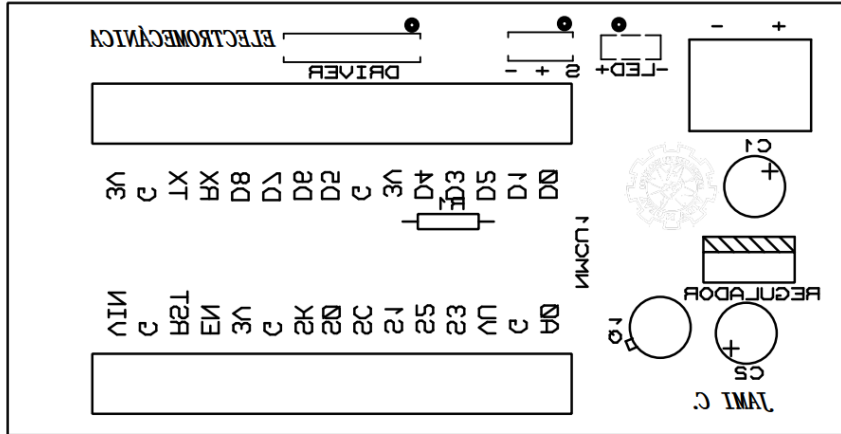


Figura 19. Importación de serigrafía en PDF

Fuente: Elaboración propia

Diseño Mecánico

La estructura del robot consta de dos partes que son la base y la estructura externa que cubre al robot incluida el soporte para las lámparas ultravioletas. Toda la parte de diseño mecánico del robot se encuentra en este apartado, se da a conocer todos los diseños realizados en 3D utilizando el Software de SolidWorks.

Diseño de la base de los elementos

Dentro de la parte mecánica se diseña una base donde apoyaran todos los componentes que conforman el prototipo. Se ha construido una base sólida en aluminio por su bajo peso y alta durabilidad en las dimensiones construidas para el robot. El aluminio es un ángulo de 90° con 25mm de ancho y 2 mm de espesor, las dimensiones del chasis son de 330 mm ancho x 450 mm de longitud y 80 mm de altura. El diseño está realizado con las dimensiones reales, se ha utilizado todas las herramientas necesarias del software para tener un acabado idéntico, como se puede apreciar a continuación.

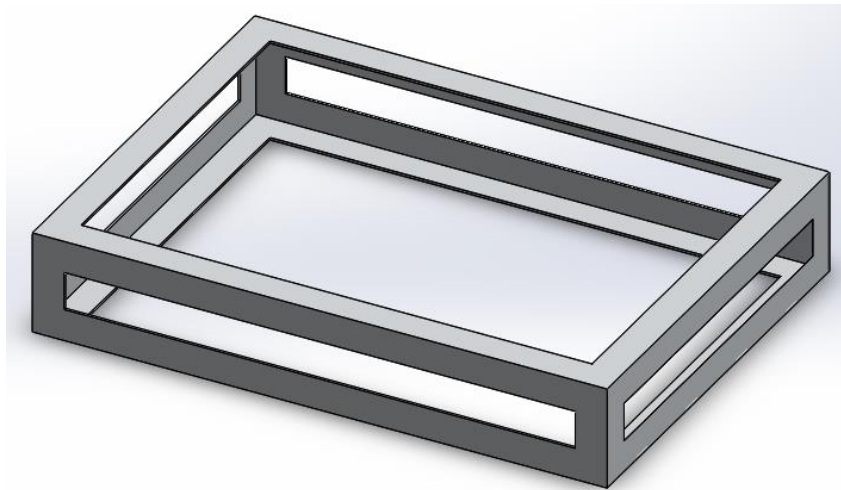


Figura 20. Diseño de la base 3D en SolidWorks

Fuente: Elaboración propia

Estructura Externa

Para la estructura del robot se realiza en el mismo software, proporcionando un acabado con el mismo material a utilizar que es de acero galvanizado. El diseño es de un prisma rectangular con chaflán en la parte delantera donde va montado los controles de mando manual y que también sirve de referencia la parte delantera del robot también cuenta con una barra metálica ubicado en la parte central de la estructura que sirve de apoyo para las lámparas y tiene una longitud de 1.3 m.

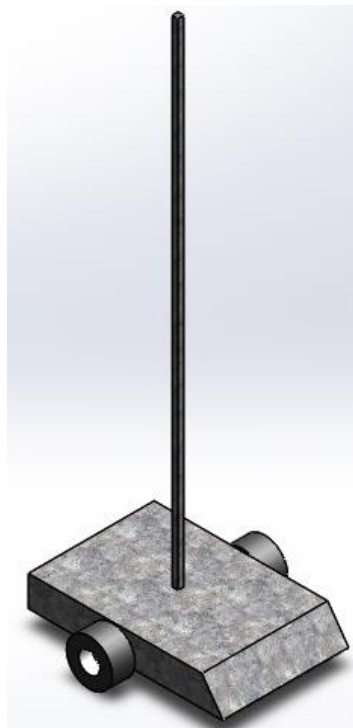


Figura 21. Diseño de la estructura 3D en SolidWorks

Fuente: Elaboración propia

El diseño para el movimiento es de un robot diferencial con ruedas en el centro lateral del robot. El diseño permite realizar giros en el mismo punto ocupando menos espacio para realizar los giros como suelen ser otros diseños que tienen las ruedas en la parte posterior. Para el equilibrio se colocó unas ruedas locas en la parte inferior, adelante y atrás del robot que permite soportar el peso que se distribuye en cuatro puntos de apoyo haciendo que el robot sea muy estable al momento de movilizarse en diferentes direcciones.

Ensamblaje eléctrico-electrónico

El ensamble eléctrico se realiza con cable AWG 12 desde la batería para alimentar a los demás componentes utilizando cable AWG 18. La conexión está realizada con terminales en U y terminales hembra y tiene una caja de conexiones donde se alimentan cada uno de los componentes, también se utilizó termo incogibles para las conexiones y en los motores y evitar contactos que puedan causar una falla eléctrica total del sistema.



Figura 22. Ensamblaje eléctrico

Fuente: Elaboración propia

En el ensamblaje electrónico se construyó la PCB soldando todos los componentes electrónicos a la placa PCB para la comunicación entre el driver y el relé se insertó para la colocación de terminales macho y su posterior conexión con cables Dupont hembra - hembra. Se realizó hoyos para poder sujetar la placa a la base del robot.

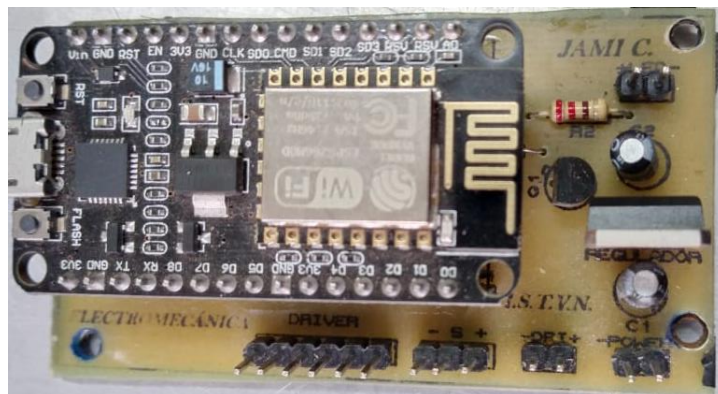


Figura 23. Ensamblaje eléctrico

Fuente: Elaboración propia

Ensamblaje Mecánico

Una vez instaladas las ruedas locas y los motores que distribuirán el peso en estos apoyos se puede observar el acabado final, el peso está distribuido adecuadamente e insertado la estructura externa que cubre al robot que es de acero galvanizado.



Figura 24. Ensamblaje del robot

Fuente: Elaboración propia

El robot ensamblado completamente y en funcionamiento de las lámparas se puede apreciar la dimensión del robot y su estética, con una excelente estabilidad al estar todo muy bien dimensionado.



Figura 25. Ensamblaje final del robot

Fuente: Elaboración propia

Programación de la placa de desarrollo

Importación de librerías

Primero se importan las librerías necesarias que la NodeMCU necesita para ejecutar las acciones que se han programado para la placa.

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
```

Figura 26. Declaración de Librerías en Arduino

Fuente: Elaboración propia

Declaración de variables

En este apartado se encuentra las variables como de los motores A y B con los pines D5, D7 y D8 para el motor A y D6, D3y D4 para el motor B para estas variables de declara con la constante #define y para las variables de velocidad y del pin que control de carga se utiliza la variable int y la variable String command donde almacena el estado del comando de la aplicación, también una variable del estado del switch del control de carga para las lámparas. Finalmente, la variable de constante char que es el nombre que se le ha asignado al servidor web de la NodeMCU.

```
#define ENA 14 // Enable/speed motors Right GPIO14 (D5)
#define ENB 12 // Enable/speed motors Left GPIO12 (D6)
#define IN_1 15 // Driver motors Right GPIO15 (D8)
#define IN_2 13 // Driver motors Right GPIO13 (D7)
#define IN_3 2 // Driver motors Left GPIO2 (D4)
#define IN_4 0 // Driver motors Left GPIO0 (D3)
#define lamp 5 // PIN DEL SW ANDROID GPIO4 (D2)

String command; //String to store app command state.
int speedCar = 200; // 400 - 1023.
int speed_Coeff = 3;

int st=0;
const char* ssid = "NodeMCU Car";
```

Figura 27. Creación de variables en Arduino

Fuente: Elaboración propia

Creación de la Función Void Setup

En esta función se declara las entradas y salidas en este caso las salidas como de los motores, control de la lámpara y la comunicación entre la NodeMCU y el dispositivo y también se crea funciones para el control de los motores.

```
void setup() {  
  
  pinMode(ENA, OUTPUT);  
  pinMode(ENB, OUTPUT);  
  pinMode(IN_1, OUTPUT);  
  pinMode(IN_2, OUTPUT);  
  pinMode(IN_3, OUTPUT);  
  pinMode(IN_4, OUTPUT);  
  
  pinMode(lamp, OUTPUT);  
  
  Serial.begin(9600);  
}
```

Figura 28. Creación de función void setup en Arduino

Fuente: Elaboración propia

Conexión Wifi

Para establecer conexión entre la NodeMCU y el dispositivo, se configura la NodeMCU en el modo punto de acceso wireless (soft-AP). El soft-AP establece un punto de acceso wireless para crear una red Wifi con el nombre de “NodeMCU Car”. Se puede agregar parámetros para la conexión como contraseña, número máximo de dispositivos conectados, canales de conexión entre otros, pero es opcional estos parámetros.

```
// Connecting WiFi  
  
WiFi.mode(WIFI_AP);  
WiFi.softAP(ssid);  
  
IPAddress myIP = WiFi.softAPIP();  
Serial.print("AP IP address: ");  
Serial.println(myIP);
```

Figura 29. Conexión wifia en Arduino

Fuente: Elaboración propia

Creación de funciones

En este apartado se encuentran las funciones creadas para el control de los motores, hacia adelante, atrás, derecha, izquierda la función empieza con Void Seguido de un nombre asignado. Para realizar movimientos como hacia delante con un ligero giro a la derecha y lo mismo para todos os sentidos se crea la Void con el nombre, para la velocidad se lo divide a otro valor obteniendo una velocidad mayor en una rueda y otra rueda menor logrando un giro controlado, como se muestra en la imagen sobre la programación.

```
void goAhead() {  
  
    digitalWrite(IN_1, LOW);  
    digitalWrite(IN_2, HIGH);  
    analogWrite(ENA, speedCar);  
  
    digitalWrite(IN_3, LOW);  
    digitalWrite(IN_4, HIGH);  
    analogWrite(ENB, speedCar);  
}  
  
void goBack() {  
  
    digitalWrite(IN_1, HIGH);  
    digitalWrite(IN_2, LOW);  
    analogWrite(ENA, speedCar);  
  
    digitalWrite(IN_3, HIGH);  
    digitalWrite(IN_4, LOW);  
    analogWrite(ENB, speedCar);  
}  
  
void goRight() {  
  
    digitalWrite(IN_1, HIGH);  
    digitalWrite(IN_2, LOW);  
    analogWrite(ENA, speedCar);  
  
    digitalWrite(IN_3, LOW);  
    digitalWrite(IN_4, HIGH);  
    analogWrite(ENB, speedCar);  
}  
  
void goLeft() {  
  
    digitalWrite(IN_1, LOW);  
    digitalWrite(IN_2, HIGH);  
    analogWrite(ENA, speedCar);  
  
    digitalWrite(IN_3, HIGH);  
    digitalWrite(IN_4, LOW);  
    analogWrite(ENB, speedCar);  
}  
  
}  
  
void goAheadLeft() {  
  
    digitalWrite(IN_1, LOW);  
    digitalWrite(IN_2, HIGH);  
    analogWrite(ENA, speedCar);  
  
    digitalWrite(IN_3, LOW);  
    digitalWrite(IN_4, HIGH);  
    analogWrite(ENB, speedCar/speed_Coeff);  
}
```

Figura 30. Creación de funciones Void en Arduino

Fuente: Elaboración propia

Creación de la función Void gosw

Esta función permite controlar el encendido y apagado de las lámparas mediante la activación del pin de la NodeMCU, el comando enviado al presionar en el icono de la app este recibe y se guarda en una constante de valor para después ser afirmada con la función.

```
void gosw() {  
  if(st==0) {  
    st=1;  
    digitalWrite(lamp,HIGH);  
  }else{  
    digitalWrite(lamp,LOW);  
    st=0;  
  }  
}
```

Figura 31. Creación de función Void gosw Arduino

Fuente: Elaboración propia

Creación de la función Void loop

En esta función se ejecuta un número infinito de veces, se crean los comandos de cada uno de los iconos de la app para el control del robot. Los comandos están asignados con letras y números, se puede cambiar los comandos y valores de la velocidad.

```
void loop() {  
  server.handleClient();  
  
  command = server.arg("State");  
  if (command == "F") goAhead();  
  else if (command == "Z") gosw();  
  else if (command == "B") goBack();  
  else if (command == "L") goLeft();  
  else if (command == "R") goRight();  
  else if (command == "I") goAheadRight();  
  else if (command == "G") goAheadLeft();  
  else if (command == "J") goBackRight();  
  else if (command == "H") goBackLeft();  
  else if (command == "0") speedCar = 200;  
  else if (command == "1") speedCar = 500;  
  else if (command == "2") speedCar = 800;  
  else if (command == "S") stopRobot();  
}
```

Figura 32. Creación de función Void Loop Arduino

Fuente: Elaboración propia

Desarrollo de la App

Para la creación de la app fue creado en la plataforma de MIT App Inventor, para acceder a la plataforma se lo realiza con una cuenta, esta plataforma se ha utilizado en anteriores proyectos por lo cual se tiene conocimiento en el desarrollo de Apps con fines educativos dentro de la plataforma.

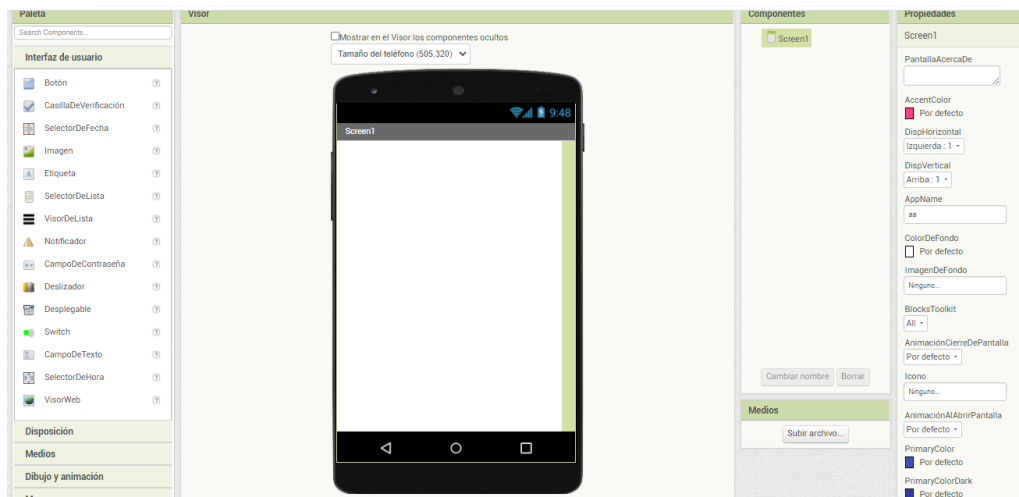


Figura 33. Pantalla principal de la App

Fuente: Elaboración propia

Antes de empezar se cuenta con imágenes que se ha insertado en el transcurso del desarrollo de la App, estas imágenes son de flechas lo cual permite identificar la dirección del robot, se utiliza varias herramientas como separadores para cada cuadro de mismo modo se puede realizar modificaciones como color, tipo de texto, tamaño entre otros.

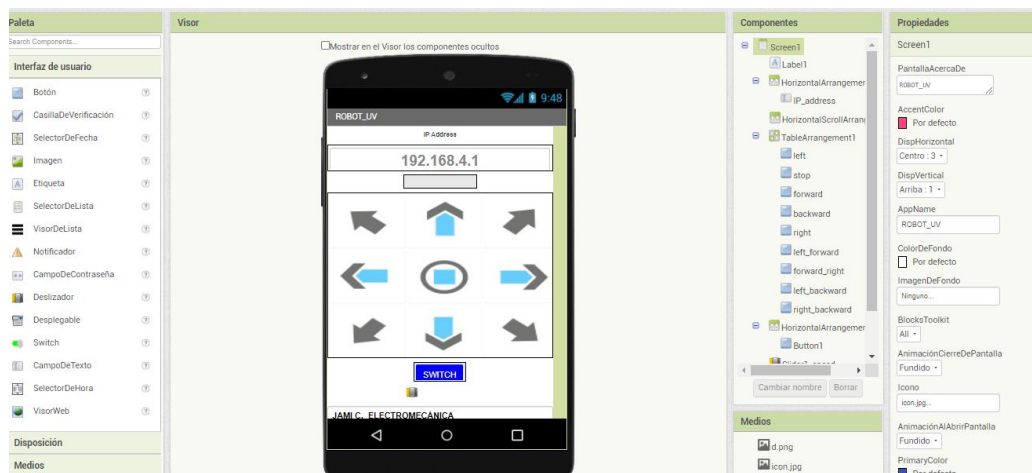


Figura 34. Desarrollo de la App

Fuente: Elaboración propia

Diagrama de bloques

Después de haber terminado el diseño se procede al diagrama de bloques donde se dará control a cada cuadro creado que son los controles del robot.

En la programación de Arduino se utilizó comandos para cada cuadro, aquí es donde se asigna los mismos comandos y la dirección IP donde se conectará la App.

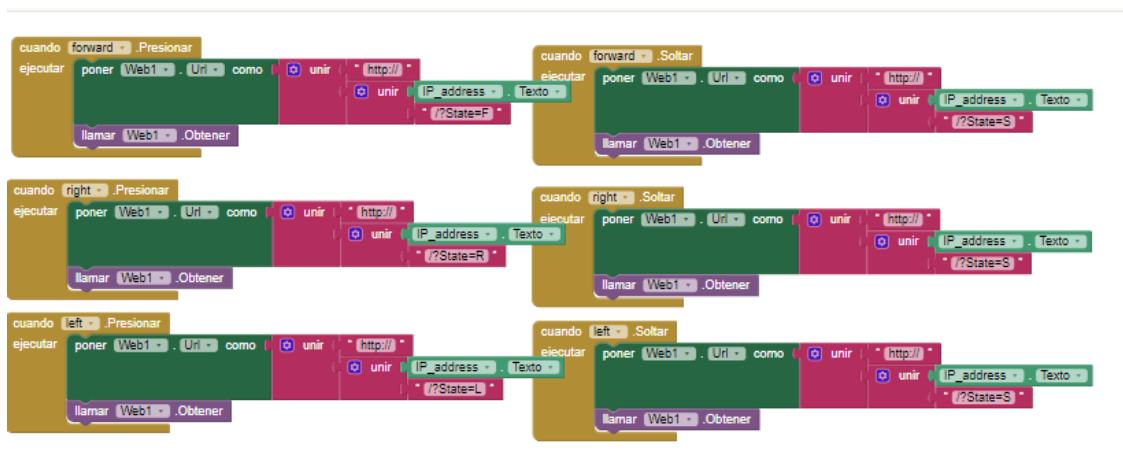


Figura 35. Diagrama de bloques

Fuente: Elaboración propia

Terminado la parte de bloques y cerciorado que todo está en orden se procede a generar la respectiva App en archivo Apk. Que posteriormente es instalado en el dispositivo móvil. Ante cualquier inconformidad del aspecto de la App se puede realizar los respectivos cambios hasta estar conforme con el aspecto de la App y descargarla nuevamente.



Figura 36. Ícono de la aplicación

Fuente: Elaboración propia

CAPÍTULO IV

RESULTADOS

El diseño del robot cumple las expectativas propuestas planteadas con una buena estabilidad gracias a distribución correcta del peso siendo su movimiento controlado y que su recorrido en espacios reducidos sea la adecuada, ocupando la misma dimensión del robot para sus giros, esto gracias a que el movimiento del robot es diferencial, que le permite esa libertad de movimiento en cualquier dirección y su velocidad también es controlado para un mejor manejo del robot, de esta forma el robot de desinfección realiza el trabajo de forma remota dentro de un área.

La desinfección por parte de la luz ultravioleta que lleva incorporado el robot tiene una eficiencia del 99.99 % en eliminación de bacterias y virus, la radiación UV con rayos tipo C (de 200 a 290 nm). La lámpara que lleva el robot es del mismo tipo de rayo y su onda oscila entre 168 y 254 nm (nanómetros). Siendo el rango suficiente para eliminar las bacterias y virus sobre cualquier superficie y mitigar la propagación del Covid- 19.

Las dimensiones finales del robot en (centímetros) sin el soporte de las lámparas ancho = 48, largo = 51 y alto = 12. El alto total incluido el soporte de las lámparas es de 143 cm. Y con un peso de 8.6 kg. Todo el peso está muy bien distribuido tal como el soporte de las lámparas que está en el centro de la estructura base, dando como resultado un excelente equilibrio. El robot cuenta con un switch principal que alimenta todo el sistema del robot, el mismo que es de tres posiciones: apagado encendido y modo de carga, cuenta con un switch únicamente para las lámparas esto le permite activar con la alimentación externa y también recargar la batería o únicamente para la carga de batería.

CONCLUSIONES

- El diseño de tracción diferencial del robot permite una buena maniobrabilidad del robot al momento de movilizarse en cualquier dirección ya que puede realizar un giro de 360 grados en su propio eje.
- La luz ultravioleta se ha convertido en un buen aliado para la desinfección ante la actual pandemia, que es muy efectivo para eliminar virus y bacterias en superficies y prevenir la propagación del Covid – 19.
- El peso del robot es esencial para dimensionar el tipo de motor adecuado. Para que el robot que tenga la suficiente tracción para mover todos sus elementos.
- La batería al ser de mayor peso tiende a ser importante para darle estabilidad y tiene una duración aproximada de 90 min y su tiempo de carga es de 120 min.
- El equilibrio depende de una base sólida con apoyos en puntos esenciales, porque el movimiento produce un balanceo mínimo por las lámparas y esto es contrarrestado por la base, evitando que el robot pierda el equilibrio y se vuelque.

RECOMENDACIONES

- Es recomendable realizar el dimensionamiento del robot previo a la adquisición de los componentes, seguido de un diseño en 3D descartar fallas en el diseño estructural.
- Se aconseja no estar cerca de la lámpara y no mirarlo fijamente cuando esté encendida, los rayos producidos pueden afectar los ojos y quemaduras leves en la piel.
- Se sugiere utilizar módulos para el circuito electrónico por la facilidad de cambio al momento de una falla del componente.

ANEXOS

Anexo 1: Código de programación del robot

```
#define ENA 14 // Enable/speed motors Right GPIO14(D5)
#define ENB 12 // Enable/speed motors Left GPIO12(D6)
#define IN_1 15 // Driver motors Right GPIO15(D8)
#define IN_2 13 // Driver motors Right GPIO13(D7)
#define IN_3 2 // Driver motors Left GPIO2(D4)
#define IN_4 0 // Driver motors Left GPIO0(D3)
#define lamp 5 // Pin del sw de Android GPIO4(D2)
```

```
String command;
int speedCar = 250; // 400 - 1023.
int speed_Coeff = 3;
```

```
int st=0;
const char* ssid = "NodeMCU Car";
ESP8266WebServer server(80);
```

```
void setup() {

  pinMode(ENA, OUTPUT);
  pinMode(ENB, OUTPUT);
  pinMode(IN_1, OUTPUT);
  pinMode(IN_2, OUTPUT);
  pinMode(IN_3, OUTPUT);
  pinMode(IN_4, OUTPUT);

  pinMode(lamp, OUTPUT);

  Serial.begin(9600);

  // Connecting WiFi

  WiFi.mode(WIFI_AP);
  WiFi.softAP(ssid);

  IPAddress myIP = WiFi.softAPIP();
  Serial.print("AP IP address: ");
  Serial.println(myIP);

  // Starting WEB-server
  server.on ( "/", HTTP_handleRoot );
  server.onNotFound ( HTTP_handleRoot );
  server.begin();
}
```

```

void goAhead(){

    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar);
}

void goBack(){

    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
}

void goRight(){

    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar);
}

void goLeft(){

    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
}

void goAheadRight(){

    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar/speed_Coeff);
}

```



```

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar);
}

void goAheadLeft(){

    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar/speed_Coeff);
}

void goBackRight(){

    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar/speed_Coeff);

    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
}

void goBackLeft(){

    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar/speed_Coeff);
}

void stopRobot(){

    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
}

```

```

void gosw(){
if(st==0){
  st=1;
  digitalWrite(lamp,HIGH);
}else{
  digitalWrite(lamp,LOW);
  st=0;
}
}

void loop() {
  server.handleClient();

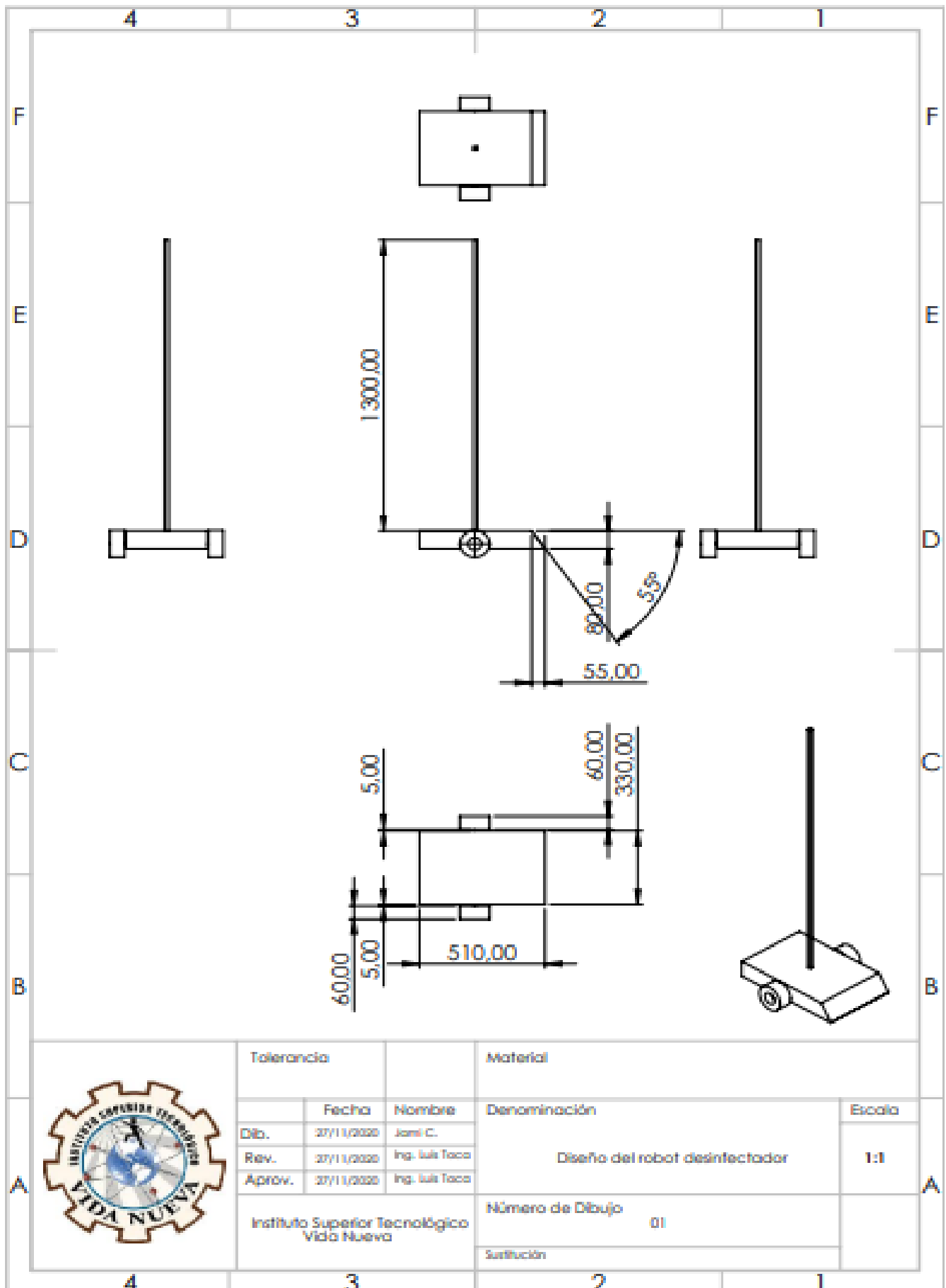
  command = server.arg("State");
  if (command == "F") goAhead();
  else if (command == "Z") gosw();
  else if (command == "B") goBack();
  else if (command == "L") goLeft();
  else if (command == "R") goRight();
  else if (command == "I") goAheadRight();
  else if (command == "G") goAheadLeft();
  else if (command == "J") goBackRight();
  else if (command == "H") goBackLeft();
  else if (command == "0") speedCar = 250;
  else if (command == "1") speedCar = 300;
  else if (command == "2") speedCar = 400;
  else if (command == "S") stopRobot();
}

void HTTP_handleRoot(void) {

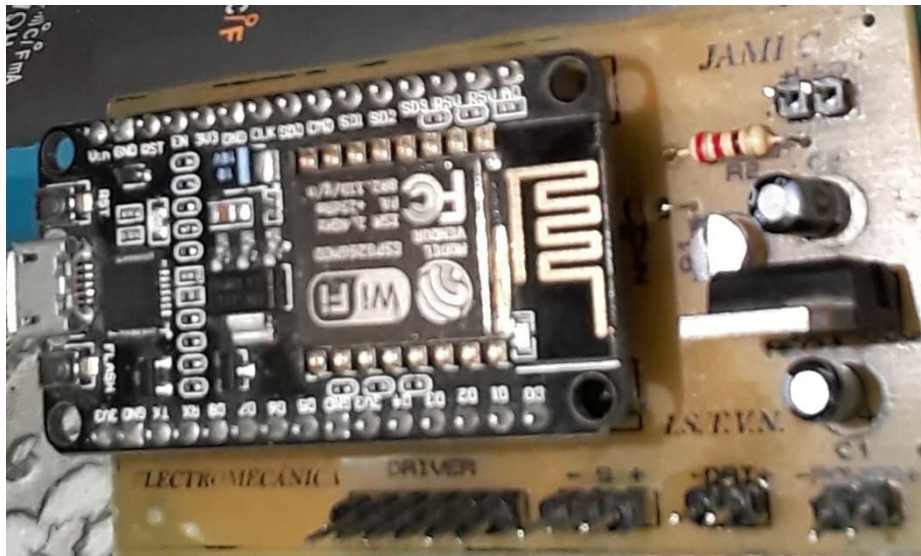
if( server.hasArg("State") ){
  Serial.println(server.arg("State"));
}
server.send ( 200, "text/html", "" );
delay(1);
}

```

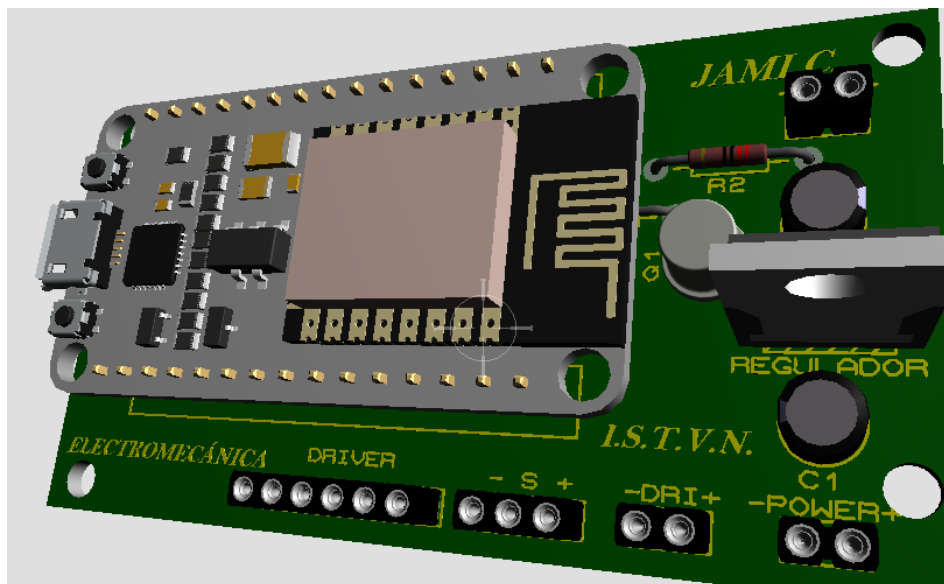
Anexo 2: Planos de la estructura



Anexo 3: PCB del circuito electrónico



Anexo 4: PCB en 3D



Bibliografía

- Anda, N. (2018). *Motor DC*. Obtenido de *Factor evolución*: <https://www.factor.mx/portal/base-de-conocimiento/motor-dc/>
- Campo, A. M. (2019). *Desinfección con luz UV, mayor eficacia en menos tiempo*. Obtenido de *Crónica Global*: https://cronicaglobal.lespanol.com/creacion/robot-xenex-desinfeccion-luz-uv_255177_102.html?fbclid=IwAR340TcyEo5Aj_RP5oX9k93p-a4ik271s147H4NJgBY20vYoqjg2leGULQ0
- Deloitte. (2020). *El impacto económico de COVID-19 (nuevo coronavirus)*. Obtenido de *Deloitte* : <https://www2.deloitte.com/ec/es/pages/strategy/articles/el-impacto-economico-de-covid-19--nuevo-coronavirus-.html#>
- Fernández, J. (2019). *¿Qué es un robot?* Obtenido de *ESPABOT*: <https://espabot.es/que-es-un-robot/>
- Fernández, J. (2020). *COVID-19: Científicos confirman que su origen es natural*. Obtenido de *Biotech*: <https://biotechmagazineandnews.com/covid-19-cientificos-confirman-que-su-origen-es-natural/>
- Figuroa, J. (2019). *360Life Technologies elige a los Robots LightStrike de Xenex*. Obtenido de *Edifarm*: https://www.edifarm.com.ec/360life-technologies-elige-a-los-robots-lightstrike-de-xenex/?fbclid=IwAR1JOa_-1btZniOCUv8VdmIFvl9RzUshKpsGOUfmq7jC5pAriNLk01ZG-Ho
- Llamas, L. (2018). *NODEMCU, LA POPULAR PLACA DE DESARROLLO CON ESP8266*. Obtenido de *LUIS LLAMAS*: <https://www.luisllamas.es/esp8266-nodemcu/>
- Marin, J. (2020). *Robots de desinfección ¿cuales son y cómo funcionan?* Obtenido de *Tuotrodiario*: <https://www.tuotrodiario.com/tecnologia/20200516202329/robots-desinfeccion-coronavirus-jam>

- MedinePlus. (2020). Síndrome respiratorio de Oriente Medio (MERS). Obtenido de MedinePlus:
<https://medlineplus.gov/spanish/ency/patientinstructions/000723.htm>*
- OMS. (2020). Concluye una histórica Asamblea de la Salud con el compromiso mundial de respuesta a la COVID-19. Obtenido de Organización Mundial de la Salud:
<https://www.who.int/es/news-room/detail/19-05-2020-historic-health-assembly-ends-with-global-commitment-to-covid-19-response>*
- OMS. (2020). Coronavirus. Obtenido de Organización Mundial de la Salud:
<https://www.who.int/es/health-topics/coronavirus>*
- Planas, O. (2016). INVERSOR DE CORRIENTE. Obtenido de Energía Solar:
<https://solar-energia.net/energia-solar-fotovoltaica/elementos/convertidores-corriente>*
- Posada, F. (2019). Creando aplicaciones para móviles Android con. Obtenido de INTEF:
<https://intef.es/wp-content/uploads/2019/03/MIT-App-Inventor-2.pdf>*
- Prieto, M. (2017). Lámparas UV-C: ¿puede algo que mata bacterias y virus ser dañino también para la salud de las personas? Obtenido de Smart Lighting:
<https://smart-lighting.es/lamparas-uv-c-salud-personas/>*
- Unicef. (2020). COVID-19: Cómo asegurar el aprendizaje de los niños sin acceso a Internet. Obtenido de Unicef Ecuador:
<https://www.unicef.org/ecuador/historias/covid-19-c%C3%B3mo-asegurar-el-aprendizaje-de-los-ni%C3%B1os-sin-acceso-internet>*
- Vazquez, A. (2020). Desarrollan robot mexicano para desinfectar hospitales a través de luz UV. Obtenido de Investigación y Desarrollo:
https://invdes.com.mx/tecnologia/desarrollan-robot-mexicano-para-desinfectar-hospitales-a-traves-de-luz-uv-video/?fbclid=IwAR1y-HORh_1Au8ZH_aL1zY5NtkITQBG1kDHCMzFjSWAtMDIIUdws0-GIxCI*