

**Escuela Superior Politécnica del Litoral**

**Facultad de Ingeniería en Mecánica y Ciencias de la Producción**

Diseño de un robot móvil recolector de bandejas de comida rápida para comedores

INGE-2308

**Proyecto Integrador**

Previo la obtención del Título de:

**Ingeniero en Mecatrónica**

Presentado por:

Juan Carlos Yépez Ortiz

Kabir Iván Cañarte Manrique

Guayaquil - Ecuador

Año: 2023

## **Dedicatoria**

---

El presente proyecto lo dedico a mis padres, Juan Carlos Yépez y Jenny Ortiz, su sacrificio y esfuerzo han sido un ejemplo a seguir en toda mi carrera universitaria. Ustedes han sido la inspiración que me ha llevado a alcanzar esta meta.

A mi querida abuelita Violeta, quien con mucho amor ha estado siempre presente en mi vida, desde mis primeros pasos. Gracias por su cariño y sabios consejos.

Y todos quienes han sido parte fundamental en este proceso, amigos más cercanos, a mi comunidad. A Andrea, por ser mi mayor cómplice, por escucharme, entenderme, animarme, y haber estado siempre, de manera incondicional, en los buenos y no tan buenos momentos. Este logro es para todos ustedes.

**Juan Carlos Yépez Ortiz**

## **Dedicatoria**

---

El presente proyecto se lo dedico a mis padres, Ing. Milton Cañarte y Lic. Nilsa Manrique, por poner en mí toda su fe y por enseñarme el valor de la superación diaria para salir adelante.

A mis hermanos, Milton Cañarte, Luisana Cañarte y Juan Pablo Cañarte, por su apoyo incondicional. Por darme siempre consejos positivos y no dejarme solo en ningún momento.

**Kabir Ivan Cañarte Manrique**

## Agradecimientos

---

Agradezco a Dios en mi primer lugar, por guiarme con sabiduría y brindarme la fortaleza en cada etapa de mi vida y en este proceso académico.

Mi más sincero agradecimiento a mis padres, Juan Carlos Yépez y Jenny Ortiz, por brindarme todas las herramientas necesarias y oportunidades que me han permitido llegar hasta este momento. Por haberme inculcado valores sólidos de ética para mi crecimiento personal y profesional, y por su incansable esfuerzo. Ustedes son mi ejemplo a seguir.

A mis tutores, Dr. Carlos Saldarriaga y Dr. Marcelo Fajardo, por su conocimiento y orientación los cuales han sido esenciales en el desarrollo de este proyecto.

Agradezco a mis compañeros con quien tuve la oportunidad de compartir esta gran etapa, especialmente a Kabir, mi compañero de tesis quien, con su amistad, ha contribuido a que este viaje sea memorable. Juntos hemos formado un gran equipo durante estos 5 años.

**Juan Carlos Yépez Ortiz**

## Agradecimientos

---

Agradezco a Dios, por haberme brindado salud y sabiduría en todo este proceso, por darme su amor en cada momento.

Quiero agradecer a mis padres, Ing. Milton Cañarte y Lic. Nilsa Manrique, cuyo amor incondicional, apoyo constante y sacrificios han sido la fuerza motriz detrás de mi viaje académico. Su aliento y confianza en mí han sido el faro que me ha guiado en cada paso. Sin su apoyo, este logro no sería posible.

También quiero agradecer a mis queridos hermanos, Milton Cañarte, Luisana Cañarte y Juan Pablo Cañarte, por ser fuentes inagotables de inspiración y motivación. Sus palabras alentadoras y su apoyo constante han sido fundamentales para superar los desafíos de este proceso.

**Kabir Ivan Cañarte Manrique**

## Declaración Expresa

---

Nosotros Juan Carlos Yépez y Kabir Cañarte acordamos y reconocemos que:

La titularidad de los derechos patrimoniales de autor (derechos de autor) del proyecto de graduación corresponderá al autor o autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores. La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de nuestra innovación, de ser el caso.

En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique los autores que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, enero del 2024.



---

Kabir Iván Cañarte  
Manrique



---

Juan Carlos Yépez Ortiz

## **Evaluadores**

---

**Efraín Terán, M.Sc.**

Profesor de Materia

---

**Carlos Saldarriaga, Ph.D.**

Tutor de proyecto

## Resumen

Este proyecto aborda el diseño integral de un robot móvil dedicado a la recolección de bandejas de comida rápida en comedores. Durante el desarrollo del proyecto, se detallaron los componentes mecánicos, destacando la creación de una estructura mecánica en forma de soporte con capacidad para albergar múltiples bandejas y soportar cargas adecuadas. Un análisis estático aseguró la robustez de la estructura y sus componentes. En el diseño electrónico, se seleccionó una batería LiPO para garantizar el rendimiento adecuado de motores y componentes electrónicos. En el ámbito computacional, se desarrolló un modelo de mapeo y navegación autónoma basado en algoritmos que posibilitaron al robot crear un mapa detallado de su entorno y determinar rutas óptimas en tiempo real, empleando un sensor LiDAR en Gazebo. También se desarrolló un sistema de reconocimiento e identificación de imágenes para localizar mesas con bandejas usadas. Utilizando OpenCV, se logró detectar un objeto indicador, permitiendo al robot ajustar su movimiento de manera autónoma. El proyecto culmina con un robot móvil eficiente y versátil, con la capacidad de navegar de forma autónoma, evitando obstáculos gracias a la información proporcionada por el LiDAR, y con la habilidad de detectar y seguir de un objeto indicador de manera precisa.

**Palabras Clave:** robot móvil, estructura mecánica, navegación autónoma, LiDAR, reconocimiento de imágenes

### ***Abstract***

*This project focuses on the integral design of a mobile robot dedicated to the collection of fast-food trays in dining areas. During the development of the project, the mechanical components were detailed, highlighting the design of a mechanical structure capable of holding multiple trays and supporting appropriate loads. A static analysis ensured the robustness of the structure and its components. In the electronic design, a LiPO battery was selected to ensure adequate performance of the motors and electronic components. In the computational area, a mapping and autonomous navigation model was developed based on algorithms that allowed the robot to create a detailed map of its environment and determine optimal routes in real time, using a LiDAR sensor in Gazebo. Additionally, a system for image recognition and identification was developed to locate tables with used trays. Using OpenCV, an indicator object was successfully detected, enabling the robot to autonomously adjust its movement. The project culminates with an efficient and versatile mobile robot, with the ability to navigate autonomously, avoiding obstacles through the information provided by the LiDAR, and with the ability to detect and follow an indicator object precisely.*

***Keywords:*** *mobile robot, mechanical structure, autonomous navigation, LiDAR, image detection*

## Índice general

Resumen .....	I
<i>Abstract</i> .....	II
Índice general .....	III
Abreviaturas .....	VI
Simbología .....	VII
Índice de figuras .....	VIII
Índice de tablas.....	X
Índice de planos.....	XI
Capítulo 1 .....	1
1.    Introducción .....	2
1.1    Descripción del problema.....	5
1.2    Justificación del problema .....	7
1.3    Objetivos.....	8
1.3.1    Objetivo general .....	8
1.3.2    Objetivos específicos.....	8
1.4    Marco teórico.....	9
1.4.1    Industria 4.0: Paradigma .....	9
1.4.2    Robots móviles autónomos (AMR).....	10
1.4.3    Percepción de un ARM: sensor LiDAR .....	11
1.4.4    ROS 2 .....	12

1.4.5	Estado del Arte .....	13
Capítulo 2 .....		17
2.	Metodología. ....	18
2.1	Selección de alternativa de solución.....	18
2.2	Proceso de diseño .....	23
2.3	Requerimientos de diseño.....	23
2.4	Diseño conceptual.....	24
2.5	Parámetros de diseño mecánico.....	25
2.5.1	Selección de motores.....	27
2.5.2	Selección de ruedas y accesorios adicionales.....	31
2.6	Parámetros de diseño electrónico .....	32
2.6.1	Selección de controlador principal .....	32
2.6.2	Selección de driver para motores .....	33
2.6.3	Selección de controlador para motores .....	35
2.6.4	Selección de cámara y sensor LiDAR.....	35
2.6.5	Consumo de corriente del robot .....	36
2.6.6	Selección de regulador .....	37
2.6.7	Selección de batería.....	37
2.7	Parámetros de diseño computacional.....	39
2.7.1	Modelo 3D del robot .....	39
2.7.2	Mapa del entorno de simulación .....	42

2.7.3	Algoritmos de navegación y localización (SLAM).....	44
2.7.4	Algoritmo de visión en OpenCV.....	45
2.7.5	Distribución general de transformadas del robot .....	46
Capítulo 3	.....	47
3.	Resultados y análisis .....	48
3.1	Diseño mecánico.....	48
3.2	Diseño computacional .....	53
3.3	Análisis de costos .....	62
Capítulo 4	.....	65
4.	Conclusiones y recomendaciones.....	66
4.1	Conclusiones.....	66
4.2	Recomendaciones .....	68
Referencias	.....	69
Apéndices	.....	73

## Abreviaturas

ESPOL	Escuela Superior Politécnica del Litoral
LiDAR	<i>Light Detection and Ranging</i>
SLAM	<i>Simultaneous Localization and Mapping</i>
PLA	Poliácido láctico
CAD	<i>Computer-Aided Design</i>
3D	Tres dimensiones
FEA	<i>Finite Element Analysis</i>
ROS	<i>Robot Operating System</i>
AMRs	<i>Autonomous Mobile Robots</i>
URDF	<i>Unified Robot Description Format</i>
AMCL	<i>Adaptive Monte Carlo Localization</i>
OpenCV	<i>Open Source Computer Vision</i>
LiPo	<i>Lithium Polymer</i>

**Simbología**

mil	Milésima de pulgada
mg	Miligramo
m	Metro
mV	Milivoltio
V	Voltio
mAh	Miliamperios-hora
A	Amperio
DC	Corriente directa
N	Newton
N.m	Newton-metro
kg	Kilogramo
kgf	Kilogramo-fuerza
s	Segundo
h	Hora
MPa	Megapascal
rpm	Revoluciones por minuto
hp	Caballos de fuerza
W	Vatio

## Índice de figuras

Figura 1.1 Robots en la industria manufacturera .....	2
Figura 1.2 Potencial de automatización por principales grupos de ocupación, 2016 .....	3
Figura 1.3 Tamaño de mercado de robots de entrega en restaurantes.....	4
Figura 1.4 Clientes de un restaurante han dejado bandejas en desorden, mostrando falta de orden y limpieza. ....	6
Figura 1.5 Tecnologías de la industria 4.0 .....	9
Figura 1.6 Robot móvil autónomo Sherpa .....	11
Figura 1.7 Pulsos láser de sensor LiDAR en un entorno de simulación .....	12
Figura 1.8 Robot móvil seguidor de línea .....	14
Figura 1.9 Robot <i>delivery</i> PuduBot.....	15
Figura 1.10 FedEx Roxo SameDay Bot.....	16
Figura 2.1 Metodología de diseño.....	23
Figura 2.2 Diseño conceptual.....	25
Figura 2.3 Vistas generales de la estructura del robot.....	26
Figura 2.4 Dimensiones del motor seleccionado .....	28
Figura 2.5 Curvas de desempeño del motor seleccionado a 12V.....	29
Figura 2.6 Sensor de efecto Hall de 2 canales.....	30
Figura 2.7 Salidas A y B del <i>encoder</i> del motor seleccionado (12 V).....	30
Figura 2.8 Rueda seleccionada.....	31
Figura 2.9 Raspberry Pi modelo 4B .....	32
Figura 2.10 Driver BTS7960 seleccionado .....	34
Figura 2.11 Batería LiPO seleccionada.....	38
Figura 2.12 Sistema de juntas y eslabones del robot: (a) Eslabón base; (b) Eslabón rueda izquierda; (c) Eslabón rueda derecha; (d) Eslabón rueda castor; (e) Eslabón base castor .....	40

Figura 2.13 Sistema de ejes coordenados del robot .....	41
Figura 2.14 Visualización del robot en Rviz .....	41
Figura 2.15 Diagrama de flujo sobre los temas y nodos del modelo 3D del robot .....	42
Figura 2.16 Vista 3D del modelo del comedor utilizado en Gazebo .....	43
Figura 2.17 Vista superior del modelo del comedor utilizado en Gazebo .....	43
Figura 2.18 Diagrama de flujo para navegación y localización .....	44
Figura 2.19 Ajuste de parámetros HSV para detección de objeto.....	46
Figura 2.20 Árbol de transformadas del robot Parte – 1 .....	47
Figura 2.21 Árbol de transformadas del robot - Parte 2.....	47
Figura 3.1 Diseño 3D del robot recolector con cargas aplicadas .....	48
Figura 3.2 Análisis de tensión de Von Mises - Soporte de motor.....	50
Figura 3.3 Análisis de tensión de Von Mises - eje de motor.....	50
Figura 3.4 Visualización del robot en Gazebo .....	53
Figura 3.5 Mapa del entorno realizado con SLAM: (a) Proceso de mapeo; (b) Resultado .....	55
Figura 3.6 Algoritmo de localización: (a) Posición inicial del robot en Gazebo; (b) Envío de pose estimada; (c) Localización de robot en pose estimada .....	57
Figura 3.7 Algoritmo de navegación: (a) Posición inicial del robot en Gazebo; (b) Envío de objetivo estimado; (c) Posición final del robot .....	58
Figura 3.8 Algoritmo de OpenCV, P1: (a) y (b) Detección de objeto; (c) Posición inicial del robot .....	60
Figura 3.9 Algoritmo de OpenCV, P2: (a) Detección de objeto; (b) Posición final del robot.....	61

## Índice de tablas

Tabla 1.1 Características principales del PuduBot.....	15
Tabla 2.1 Definición de criterios de diseño.....	19
Tabla 2.2 Alternativas de solución.....	20
Tabla 2.3 Criterios de diseño.....	21
Tabla 2.4 Matriz de decisión de alternativas de diseño.....	22
Tabla 2.5 Requerimientos de diseño .....	24
Tabla 2.6 Especificaciones de motores seleccionados .....	27
Tabla 2.7 Especificaciones de ruedas seleccionadas.....	31
Tabla 2.8 Especificaciones técnicas del controlador seleccionado (Raspberry Pi modelo 4B) ....	33
Tabla 2.9 Especificaciones del driver BTS7960 seleccionado.....	34
Tabla 2.10 Especificaciones de la cámara seleccionada .....	35
Tabla 2.11 Especificaciones del sensor LiDAR seleccionado .....	36
Tabla 2.12 Consumo de corriente de componentes de 5V .....	36
Tabla 2.13 Especificaciones del regulador de 5V seleccionado .....	37
Tabla 2.14 Especificaciones de la batería seleccionada .....	38
Tabla 3.1 Peso de alimentos en una dieta de una persona natural.....	49
Tabla 3.2 Propiedades mecánicas del aluminio.....	51
Tabla 3.3 Propiedades mecánicas del acero .....	51
Tabla 3.4 Tabla de costos de los componentes del robot .....	63

## Índice de planos

PLANO 1 Robot recolector - partes

PLANO 2 Base del chasis

PLANO 3 Parte superior de chasis

PLANO 4 Soporte para cámara Pi

PLANO 5 Unión chasis - bandeja

PLANO 6 Soporte para bandeja

PLANO 7 Unión bandeja - bandeja

PLANO 8 Soporte cabeza - bandeja

PLANO 9 Unión bandeja - cabeza

PLANO 10 Soporte para pantalla

PLANO 11 Soporte motor - chasis

# Capítulo 1

## 1. Introducción

En los últimos años, la robótica ha logrado avances notables en diversos campos, consolidándose como un pilar de la modernización [1]. Dicho impacto se extiende desde la manufactura hasta la medicina y la agricultura. La **Figura 1.1** evidencia que, en la industria manufacturera, los robots desempeñan labores monótonas y riesgosas, mejorando la calidad y eficiencia de la producción. En el sector médico, los robots quirúrgicos posibilitan procedimientos más precisos y menos invasivos. Asimismo, en la agricultura, la automatización eleva la productividad en la recolección y cuidado de cultivos [2].

**Figura 1.1**

*Robots en la industria manufacturera* [3]



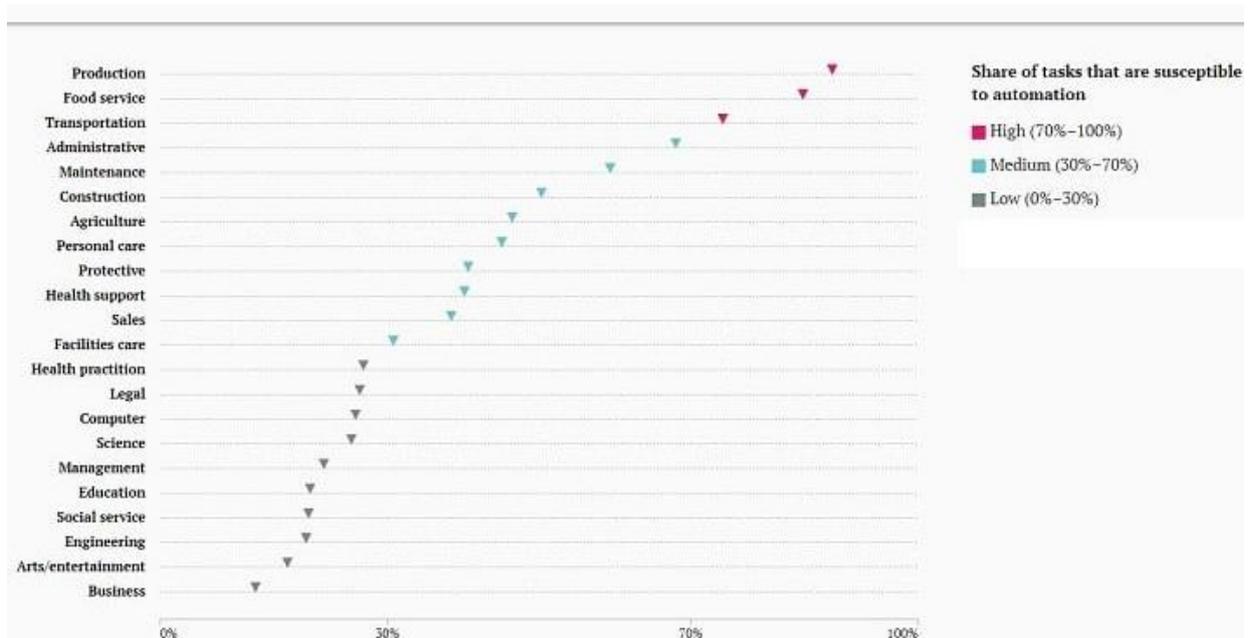
Este constante avance en la robótica demuestra la búsqueda continua de optimización y modernización en diferentes ámbitos. El “potencial de automatización de grandes grupos ocupacionales” se refiere a la evaluación del grado de vulnerabilidad de diferentes ocupaciones a la automatización a través de tecnologías como la robótica y la inteligencia artificial [4]. Esta

evaluación se basa en factores como la naturaleza de la tarea, la frecuencia de repetición y la necesidad de habilidades cognitivas y emocionales.

En 2016 se realizaron varios estudios y análisis para estimar esta posibilidad. Si bien algunos trabajos con tareas rutinarias y predecibles, como el trabajo de producción en una fábrica, se consideraron altamente automatizables, los trabajos que requieren habilidades creativas, interpersonales y de toma de decisiones se consideraron menos susceptibles a la automatización. Estos análisis tienen como objetivo comprender cómo la tecnología puede afectar la fuerza laboral y cómo abordar los cambios en el empleo en diferentes sectores, como se observa en la **Figura 1.2**.

**Figura 1.2**

*Potencial de automatización por principales grupos de ocupación, 2016 [4]*



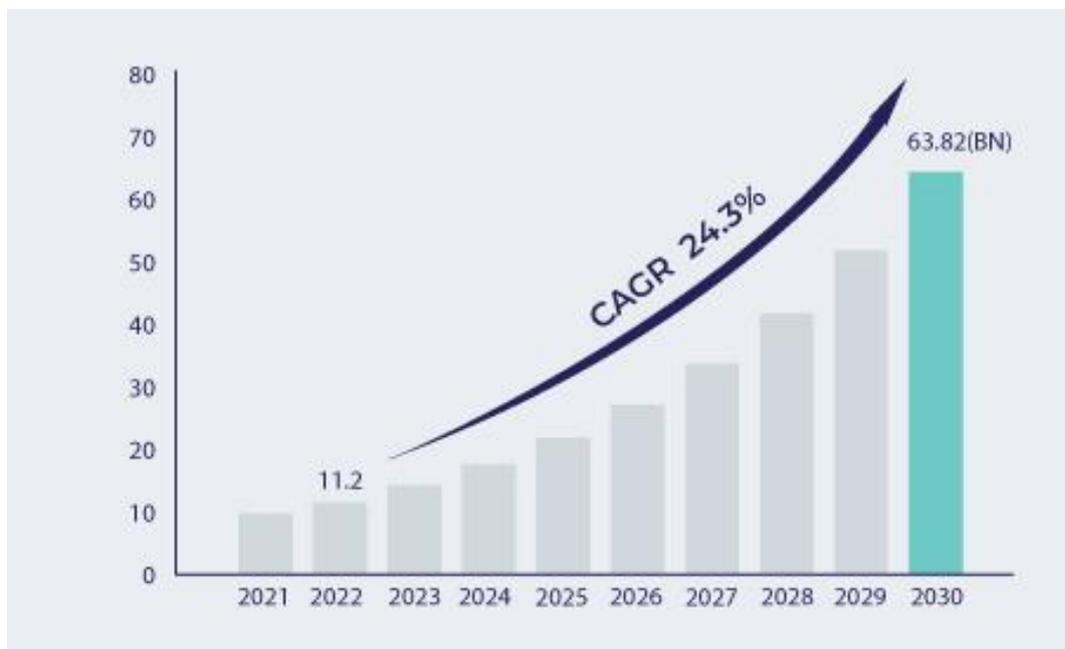
*Nota.* La figura muestra el porcentaje de los grupos de ocupación susceptibles a la automatización.

En los últimos años se han producido cambios sin precedentes en la industria de los restaurantes y la comida rápida. La pandemia de COVID-19 fue un importante punto de inflexión, pues los cierres y medidas cautelares han causado importantes pérdidas a la industria [5]. Es por

esto, que la necesidad de servicios sin contacto ha impulsado la adopción de la robótica, que se ha convertido en una herramienta para la prestación segura de servicios remotos. Por otro lado, los robots ofrecen importantes ventajas de costes y productividad, lo que los convierte en una herramienta valiosa para aumentar la competitividad y la eficiencia en la industria alimentaria [6]. Como se observa en la **Figura 1.3**, los restaurantes en particular están utilizando robots para aumentar la productividad y mejorar la experiencia del cliente al agilizar la preparación de alimentos, la entrega de pedidos y el servicio al cliente[7].

**Figura 1.3**

*Tamaño de mercado de robots de entrega en restaurantes[8]*



La investigación sobre el uso de robots en la industria de alimentos y bebidas es de gran importancia debido a la gran demanda de trabajadores, la importancia de estos empleos en la industria hotelera y la alta tasa de rotación [9]. Es así, como en un estudio realizado por la revista *Tourism Futures* en el año 2023 a 1579 personas sobre tareas en restaurantes que los clientes consideran apropiadas para la robótica, se encontró que los participantes estaban dispuestos a que un robot se hiciera cargo de su pedido en el restaurante, con una puntuación media de 5,37 en la

escala de calificación. En segundo lugar, la puntuación media de la tarea de limpieza de la mesa fue de 5,19. Finalmente, los encuestados expresaron su apoyo a la idea de que robots entreguen alimentos y bebidas, con una puntuación media de 5,16 [10].

El uso de robots en restaurantes va más allá de mejorar la percepción de los clientes sobre la calidad del servicio, pues influirá en cómo los turistas ven el destino en general. Esta percepción puede estar relacionada con aspectos como la higiene y eficiencia, que son esenciales para los clientes. Así, la experiencia positiva o negativa de los turistas con los robots en los restaurantes tendrá un impacto considerable en dichos establecimientos. [11]

### **1.1 Descripción del problema**

En la actualidad, el orden, la limpieza y mantenimiento de las mesas de los comedores han adquirido una importancia significativa en la industria del servicio de alimentos. Dada la creciente conciencia de los consumidores en relación con la seguridad e inocuidad alimentaria, los establecimientos gastronómicos se han visto obligados a primar la impecabilidad de las áreas y herramientas de servicio de comida. Tales como mesas, bandejas y alrededores, a través de la cultura de limpieza dictada sobre los empleados y equipo de trabajo.[12]

El primer aspecto que capta la atención de un individuo al ingresar a un comedor es la limpieza del entorno, dado que a partir de este criterio se infiere la calidad de la oferta gastronómica, la ética laboral del personal y el compromiso del restaurante con su clientela.

Por ende, para atender a esta problemática, resulta imperativo cumplir con ciertas demandas por parte del cliente, tales como: mantener la pulcritud de las mesas durante la entrada y salida de los comensales, preservar sistemáticamente el orden en el establecimiento, y asegurar la inocuidad y excelencia de los productos alimenticios.

**Figura 1.4**

*Clientes de un restaurante han dejado bandejas en desorden, mostrando falta de orden y limpieza.*



La concepción de este proyecto surge a partir de un problema de limpieza en un restaurante localizado en ESPOL. El establecimiento representa uno de los sitios de comida más concurridos por los estudiantes, siendo habitual su ocupación durante las horas del almuerzo. No obstante, se ha evidenciado una deficiencia en el mantenimiento de la limpieza de las mesas, de modo que los grupos de estudiantes que acuden a almorzar suelen encontrarse con residuos de comida en ellas, junto con bandejas de alimentos sin retirar.

## 1.2 Justificación del problema

Para solucionar la problemática expuesta y detallada previamente, se requiere desarrollar un robot capaz de recoger bandejas de comida, utilizando una combinación de cámara y sensor LiDAR para visualizar el entorno y orientarse hacia la mesa designada. Es importante resaltar la necesidad de abordar esta situación, dado que representaría un avance tecnológico significativo en el sector alimenticio, además de contribuir a mejorar la eficacia de la limpieza en establecimientos de comida no sólo en ESPOL, sino a nivel global. Por consiguiente, se presenta una iniciativa orientada a avanzar en la mejora continua de los restaurantes, aprovechando así las posibilidades que brinda la era de la inteligencia artificial y la industria 4.0.

El diseño de este prototipo dentro de un restaurante ayuda a que este se mantenga limpio en todo momento, ya que se trata de un robot de respuesta rápida, cada vez que se necesite limpiar alguna mesa para un nuevo comensal, lo realizará de forma inmediata, gracias al entrenamiento realizado por visión artificial y aprendizaje automático de código abierto.

Asimismo, el robot tiene como característica principal, la segmentación de imágenes. Esta herramienta permite identificar imágenes de interés, en forma de objetos o estructuras, y separar el fondo de la imagen principal. Esto ayudará a que el robot sea capaz de reconocer mesas, sillas, paredes, platos y bandejas, y mediante este reconocimiento de patrones en movimiento, podrá crear la ruta óptima para cumplir su objetivo que será recoger los desechos de cualquier mesa que necesite sanitación.

## 1.3 Objetivos

### 1.3.1 *Objetivo general*

Diseñar un robot móvil recolector de bandejas de comida rápida mediante un sistema mecatrónico y utilizando ROS2, para mejorar la eficiencia en la limpieza y recolección de basura en comedores.

### 1.3.2 *Objetivos específicos*

1. Desarrollar y validar un modelo de mapeo y navegación autónoma para el robot móvil utilizando ROS2 y un sensor LiDAR en un ambiente virtual.
2. Implementar un sistema de reconocimiento e identificación de imágenes para localizar mesas con bandejas usadas utilizando una cámara y la plataforma OpenCV.
3. Diseñar una estructura mecánica adaptativa en forma de soporte para facilitar la recolección eficiente de bandejas y otros utensilios usados.
4. Realizar pruebas para evaluar la eficacia, eficiencia y viabilidad del robot móvil en un entorno simulado utilizando Gazebo.

## 1.4 Marco teórico

### 1.4.1 *Industria 4.0: Paradigma*

La Industria 4.0 representa un concepto global, así como las nuevas tendencias en la manufactura y sectores relacionados, basados en la integración de tecnologías avanzadas. El objetivo de este concepto es recopilar y aplicar de forma inteligente datos e información en tiempo real mediante la interconexión de elementos individuales [13]. El objetivo es reducir la complejidad operativa, mejorar la eficiencia y la eficacia y reducir los costos a largo plazo.

Varias áreas tecnológicas forman parte de la Industria 4.0, las cuales son; la integración horizontal y vertical de sistemas, Internet de las cosas, ciberseguridad, nube, análisis de *big data*, simulación, fabricación aditiva (impresión 3D), realidad aumentada y robótica. Estas tecnologías redefinen los límites entre los sectores económico e industrial y entre compradores y vendedores. Los sistemas de producción actuales son cada vez más dinámicos, flexibles, eficientes y sostenibles, ofreciendo más opciones de adaptación e individualización. El impacto de las tecnologías de la Industria 4.0 será profundo y transformador en muchos sentidos [14].

#### **Figura 1.5**

*Tecnologías de la industria 4.0* [15]



#### 1.4.2 Robots móviles autónomos (AMR)

En simples términos, un robot autónomo es “una máquina que opera en un entorno parcialmente desconocido e impredecible”[16]. Los robots autónomos son parte de la generación de la industria 4.0. Estos robots están diseñados para realizar tareas en las cuales la intervención humana es mínima. Se caracterizan por estar equipados con sensores, inteligencia artificial, *software* de última generación, y ser capaces de realizar tareas difíciles y delicadas. Un ejemplo se muestra en la **Figura 1.6**.

**Figura 1.6**

*Robot móvil autónomo Sherpa [17]*



Los robots móviles autónomos (AMR) se destacan por su capacidad de tomar decisiones independientes y realizar correcciones por sí mismos. Un robot completamente autónomo tiene la habilidad de observar su entorno, realizar valoraciones basadas en la información visual o en lo que ha aprendido, y luego llevar a cabo acciones o manipulaciones en ese entorno de manera autónoma, sin requerir dispositivos de guía o control centralizado. Esta autonomía les confiere una ventaja significativa en comparación con los vehículos automatizados guiados, ya que son capaces de navegar en su entorno de forma independiente, sin depender de asistencia externa [18].

### ***1.4.3 Percepción de un ARM: sensor LiDAR***

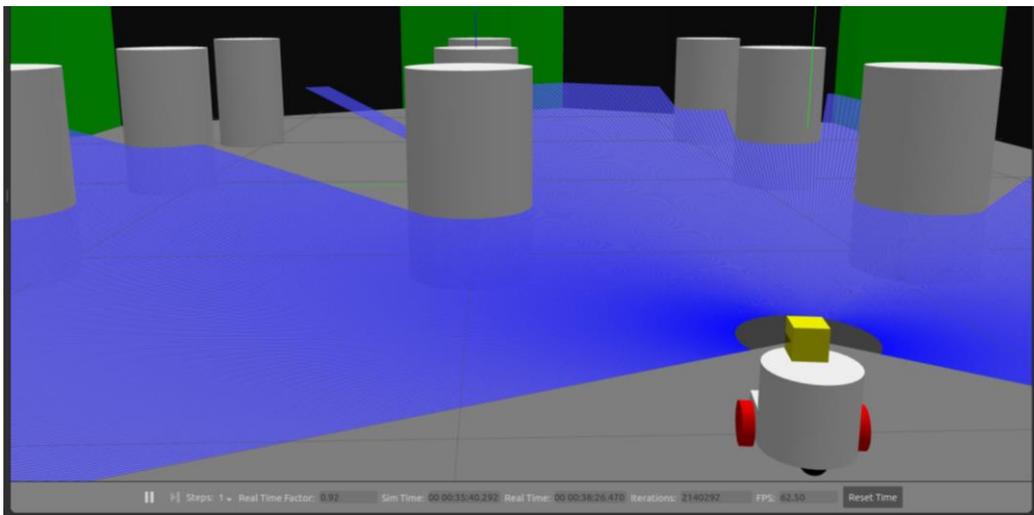
La planificación de la trayectoria es una de las tareas más importantes en el movimiento o navegación de los *AMRs*, la cual, tiene como objetivo principal, hacer que el robot encuentre una ruta óptima considerando factores de rendimiento como el tiempo de recorrido y mejor eficiencia en el consumo de energía. Para ello, se realizan mediciones con diversos sensores y se extrae

información significativa de esas mediciones [19].

Los sensores LiDAR (*Light Detection and Ranging*) son dispositivos que utilizan pulsos láser para medir la distancia y crear un mapa tridimensional del entorno. Se utiliza para el reconocimiento y la navegación de robots móviles autónomos (AMR). LiDAR escanea el entorno, recopila datos en tiempo real y crea mapas detallados que permiten a AMR identificar obstáculos, calcular distancias y planificar rutas seguras y eficientes [20].

**Figura 1.7**

*Pulsos láser de sensor LiDAR en un entorno de simulación [21]*



#### **1.4.4 ROS 2**

Para que la comunidad robótica pueda verificar y probar una propuesta de manera sencilla y efectiva, es necesario crear un entorno de software compartido y estandarizado. ROS 2 (*Robot Operating System 2*) es un sistema operativo y un kit de desarrollo de software de código abierto diseñado para aplicaciones robóticas. Su objetivo principal es proporcionar una plataforma estándar a desarrolladores de diversos sectores, facilitando la creación, control y comunicación de robots autónomos en aplicaciones que abarcan desde la investigación y la creación de prototipos hasta la producción. ROS 2 se compone de nodos independientes que se comunican mediante una

plataforma de mensajería de publicación/suscripción, lo que permite a los desarrolladores utilizar bibliotecas y herramientas para crear aplicaciones robóticas de manera eficiente y modular [22].

En el contexto de ROS, los nodos son unidades de procesamiento independientes que operan de manera autónoma. Cada nodo tiene un nombre, que registra con el maestro ROS antes de realizar cualquier otra acción. La comunicación entre nodos se basa en dos mecanismos fundamentales: temas (*topics*) y servicios (*services*).

**Temas:** Los temas son canales de comunicación asíncrona que pueden incluir desde la obtención de datos de sensores hasta comandos de actuadores, es decir, se usan para flujos de datos continuos. Un nodo publica mensajes en un tema, y otros nodos pueden suscribirse a ese tema para recibir los mensajes.

**Servicios:** Los servicios permiten la comunicación síncrona entre dos nodos y se usan para llamadas a procedimientos remotos que finalizan inmediatamente, como la captura de una imagen de un solo fotograma. Un nodo envía una solicitud a otro nodo y espera una respuesta [23].

#### ***1.4.5 Estado del Arte***

Dentro del sector de alimentos, los robots se emplean principalmente en labores relacionadas con la manipulación de productos alimentarios, el envasado, el apilamiento y diversas aplicaciones de servicios de comidas. La limpieza y recolección de alimentos constituye una de las funciones incluidas en los servicios de comida de los robots en la industria alimentaria.

Existen varias investigaciones en la aplicación de la técnica de "seguimiento de líneas" para la navegación de robots de servicio en China y Japón. En esencia, esta táctica requiere que los robots sigan una línea trazada en el suelo o una ruta predeterminada, lo que les permite moverse de manera autónoma por áreas específicas.

En este escenario, los robots están programados para identificar y seguir la línea marcada en el suelo utilizando sensores y sistemas de visión para mantenerse en el curso correcto. La navegación basada en el concepto de "seguimiento de líneas" se considera un componente esencial del desarrollo de robots autónomos porque proporciona una forma precisa y eficaz de desplazarse en entornos [24].

**Figura 1.8**

*Robot móvil seguidor de línea [25]*



Otro de los robots móviles utilizados en la actualidad es el PuduBot (**Figura 1.9**), un robot de entrega y servicio diseñado para operar en entornos como restaurantes, tiendas, supermercados, hoteles, hospitales y demás [26]. Este robot diseñado para realizar tareas como entregar alimentos o suministros a clientes en entornos comerciales. Está equipado con tecnología de navegación autónoma y capacidades de interacción con los clientes. Además, cuenta con 2 sensores LiDAR

que le proporcionan una visión constante y precisa de su entorno, lo que le permite realizar ajustes de dirección de manera intuitiva.

**Figura 1.9**

*Robot delivery PuduBot*



En la siguiente tabla se muestran las características principales del PuduBot.

**Tabla 1.1**

*Características principales del PuduBot*

<b>Capacidad de carga</b>	13 kg/bandeja, máximo 30 kg
<b>Velocidad lineal</b>	0.5-1.2 m/s
<b>Material</b>	ABS
<b>Duración de la batería</b>	10-24 horas
<b>Dimensiones</b>	51.6 x 50 x 128.8 cm

Similar al PuduBot, se encuentra otro robot llamado *FedEx Roxo SameDay Bot* (**Figura 1.10**), el cual es también un robot de entrega autónomo diseñado para realizar entregas locales en entornos urbanos. Estos robots pertenecen a una categoría denominada ADR (*Autonomous Delivery Robots*), es decir, son robots que ofrecen servicios de entrega. Según [27], se los puede clasificar según su tipo de carga en pequeños, medianos, y grandes. FedEx pertenece a los ADR medianos, los cuales pueden moverse por carreteras locales sin causar inconvenientes en el tráfico general, sin embargo, este robot específicamente cuenta con restricciones en carreteras locales debido a las normas de tráfico. No obstante, están permitidos para circular por veredas o carriles de bicicleta a bajas velocidades.

**Figura 1.10**

*FedEx Roxo SameDay Bot*



## **Capítulo 2**

## **2. Metodología.**

En el presente capítulo, se exploran diversas soluciones para abordar el problema mencionado en el capítulo anterior. Se determina una solución a desarrollar basándose en requerimientos del diseño y criterios de selección. A continuación, se ofrece una descripción más detallada de esta solución, centrándose en los componentes de las áreas mecánica, electrónica y computacional que forman parte del diseño mecatrónico.

### **2.1 Selección de alternativa de solución**

Para elegir la alternativa en función de las necesidades del cliente, se llevó a cabo un análisis empleando los parámetros de diseño indicados en la **Tabla 2.1**

**Tabla 2.1***Definición de criterios de diseño*

<b>Ligereza</b>	El diseño debe priorizar la ligereza de la estructura para ser más ágil y maniobrar de manera más efectiva en espacios estrechos y congestionados.
<b>Costo</b>	El diseño debe ser económicamente viable, se debe priorizar los costos iniciales de implementación, así como los costos de mantenimiento a largo plazo.
<b>Eficiencia</b>	El robot debe ser capaz de identificar, de manera eficiente, las mesas cuyos clientes hayan terminado su comida, en el menor tiempo posible.
<b>Robustez</b>	En su navegación, el robot debe ser capaz de evitar obstáculos adaptándose al entorno donde se encuentra.
<b>Integración</b>	Debe integrarse con sensores y software de código abierto para permitir operaciones autónomas y capacidades de localización, mapeo y navegación.
<b>Escalabilidad</b>	El robot se debe poder adaptar y expandir según las necesidades cambiantes del entorno de comida rápida, en una variedad de escenarios y cantidad de bandejas.
<b>Estética</b>	Se debe considerar la estética del robot recolector con relación a su impacto en la percepción del cliente y la imagen del restaurante.

A continuación, en la **Tabla 2.2**, se presentan 3 alternativas de solución desarrolladas en respuesta a los requerimientos de diseño previamente establecidos.

**Tabla 2.2***Alternativas de solución*

<b>Alternativa 1</b>	Diseño de un robot seguidor de línea con brazos recolectores. Se propone un robot equipado con una cámara que sigue una línea de referencia en el suelo del restaurante. Cuando detecta una bandeja con platos vacíos o semi vacíos, el robot utiliza sus brazos mecánicos con un dispositivo de sujeción para recogerla y depositarla en una plataforma de almacenamiento.
<b>Alternativa 2</b>	Diseño de un robot colaborativo con operador remoto. Se plantea un enfoque colaborativo donde un robot es controlado de forma remota por un operador humano. El operador utiliza una cámara general del restaurante y envía al robot las coordenadas de la mesa que haya finalizado para que éste se dirija a dicha mesa. El robot cuenta con una estructura mecánica en donde la persona depositará la bandeja de comida una vez que el robot haya llegado a su destino.
<b>Alternativa 3</b>	Diseño de un robot autónomo con software de mapeo y navegación. Se propone un robot autónomo equipado con un sensor LiDAR para mapeo y localización precisa. El robot utiliza software de planificación de rutas y evasión de obstáculos para operar de manera autónoma, recolectando bandejas según un cambio de estado. Cuando el robot detecte, por medio de una cámara, un indicador en cuya mesa haya finalizado de comer, se dirigirá y detendrá para que la persona deposite la bandeja de comida.

En la siguiente **Tabla 2.3** se muestra una comparación de los diferentes criterios de evaluación entre sí, indicando si un criterio es más importante que otro en una escala relativa, para conocer la prioridad y el peso de cada uno de ellos.

Tabla 2.3

Criterios de diseño

Costo=Integración>Ligereza=Eficiencia>Escalabilidad>Robustez>Estética									
Criterio	Ligereza	Costo	Eficiencia	Robustez	Integración	Escalabilidad	Estética	$\Sigma+1$	Ponderación
<b>Ligereza</b>		1	0.5	0	1	0.5	0	4	0.16
<b>Costo</b>	1		0.5	0.5	0.5	0.5	0.5	4.5	0.18
<b>Eficiencia</b>	0.5	0.5		0.5	1	0.5	0	4	0.16
<b>Robustez</b>	0	0.5	0.5		0.5	0.5	0	3	0.12
<b>Integración</b>	1	0.5	1	0.5		0.5	0	4.5	0.18
<b>Escalabilidad</b>	0.5	0.5	0.5	0.5	0.5		0	3.5	0.14
<b>Estética</b>	0	0.5	0	0	0	0		1.5	0.06
							<b>Suma</b>	25	1.00

En esta matriz, los valores "1" indican que un criterio es más importante que otro, "0.5" indica que ambos criterios son igualmente importantes, y "0" indica que un criterio es poco importante en comparación con el otro.

A continuación, se procede a realizar una matriz de decisión con el propósito de evaluar y determinar cuál de las alternativas de solución es la más adecuada y beneficiosa para el proyecto.

Los resultados se muestran en la **Tabla 2.4**.

**Tabla 2.4***Matriz de decisión de alternativas de diseño*

Alternativa 3 > Alternativa 2 > Alternativa 1									
<b>Conclusión</b>	<b>Ligereza</b>	<b>Costo</b>	<b>Eficiencia</b>	<b>Robustez</b>	<b>Integración</b>	<b>Escalabilidad</b>	<b>Estética</b>	<b><math>\Sigma+1</math></b>	<b>Prioridad</b>
<b>Alternativa 1</b>	0.034	0.045	0.0457	0.034	0.03	0.028	0.0257	0.243	3
<b>Alternativa 2</b>	0.057	0.06	0.0914	0.034	0.06	0.056	0.0171	0.376	2
<b>Alternativa 3</b>	0.069	0.075	0.0229	0.051	0.09	0.056	0.0171	0.381	1

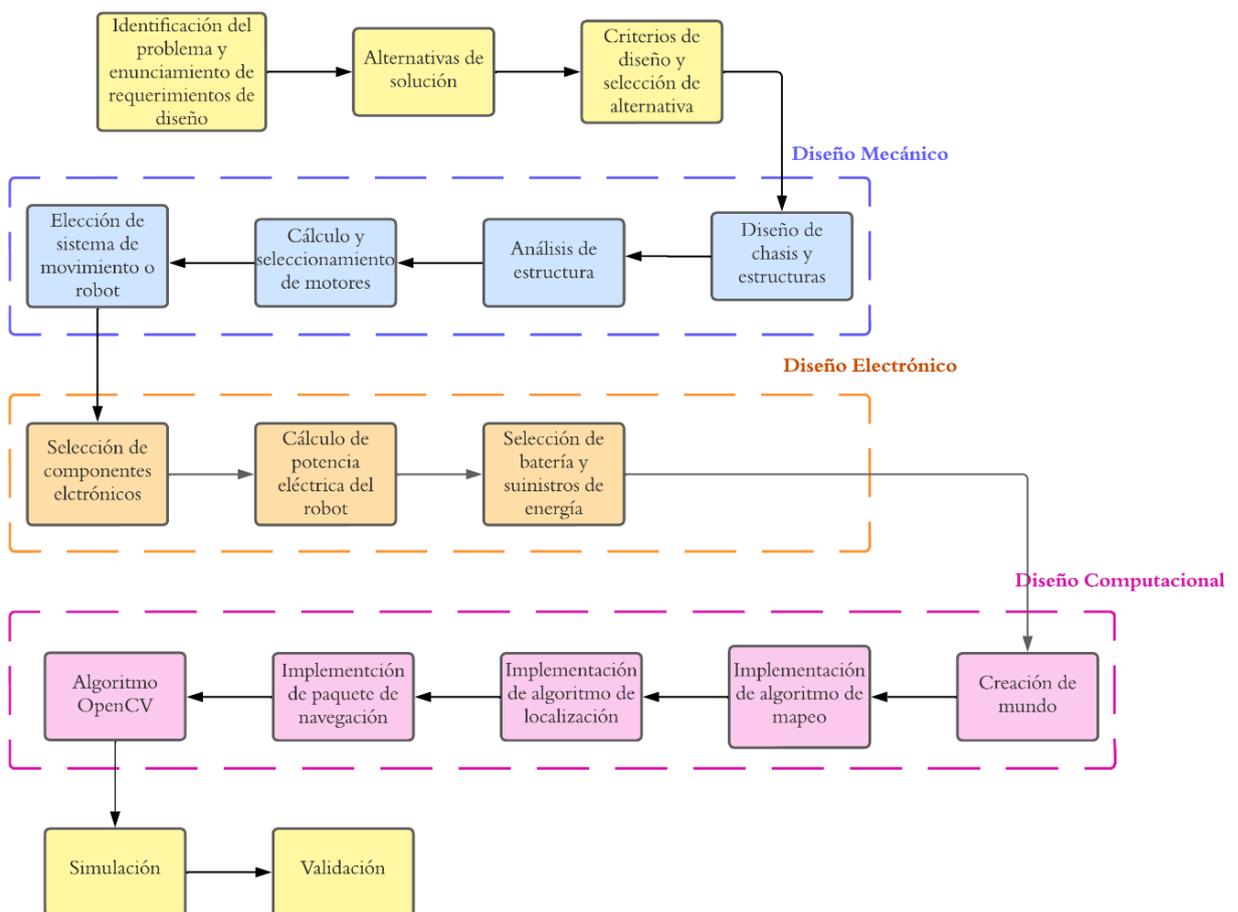
Luego de evaluar las alternativas, se obtuvo la alternativa 3 como la solución óptima para cumplir con los requisitos esenciales del proyecto. Se destacó por su ligereza, lo que garantiza agilidad y maniobrabilidad en el entorno del restaurante. Además, su eficiencia en la identificación de las mesas que han finalizado su comida y su capacidad de adaptarse a obstáculos en el entorno. A nivel de costos, esta alternativa ofreció una solución económica en términos de costos iniciales de implementación, lo que resulta fundamental para la viabilidad del proyecto. Adicionalmente, su capacidad de integración con sensores (LiDAR), cámara, y software de código abierto, permitirá operaciones autónomas y una eficiente localización, mapeo y navegación en el restaurante.

## 2.2 Proceso de diseño

Para desarrollar el proyecto, se realizó un diagrama de flujo detallado del proceso de diseño utilizado. En la **Figura 2.1** se desglosan las diversas etapas que marcaron el desarrollo del proyecto.

**Figura 2.1**

*Metodología de diseño*



## 2.3 Requerimientos de diseño

Para el diseño de un robot móvil recolector de bandejas de comida en comedores, es crucial tener en cuenta una serie de requerimientos específicos. Estos requisitos, que se detallan en la

**Tabla 2.5**, son esenciales para garantizar de que el robot cumpla con las normativas y necesidades del restaurante de manera efectiva.

**Tabla 2.5**

*Requerimientos de diseño*

<b>Capacidad de almacenamiento</b>	El robot debe permitir almacenar de 3 a 5 bandejas de comida inicialmente.
<b>Sistema de mapeo y localización</b>	El robot debe estar equipado con un sensor LiDAR capaz de realizar un mapeo preciso del entorno.
<b>Tamaño y almacenamiento</b>	Para su primera versión, el robot tendrá una dimensión de 0.30x0.40x1 m aproximadamente.
<b>Costos</b>	La inversión para la futura fabricación e implementación deberá ser la más económica posible, sin dejar de lado la eficiencia.
<b>Sistema de cambio de estado</b>	El robot debe contar con un sistema que permita cambiar su estado operativo cuando detecta un indicador de una mesa en la que ha finalizado de comer.
<b>Software de planificación de rutas</b>	El robot debe contar con un software de planificación de rutas que le permita determinar la trayectoria más eficiente y evitar obstáculos.
<b>Detección</b>	El robot debe estar equipado con una cámara capaz de detectar indicadores en las mesas de los comensales.

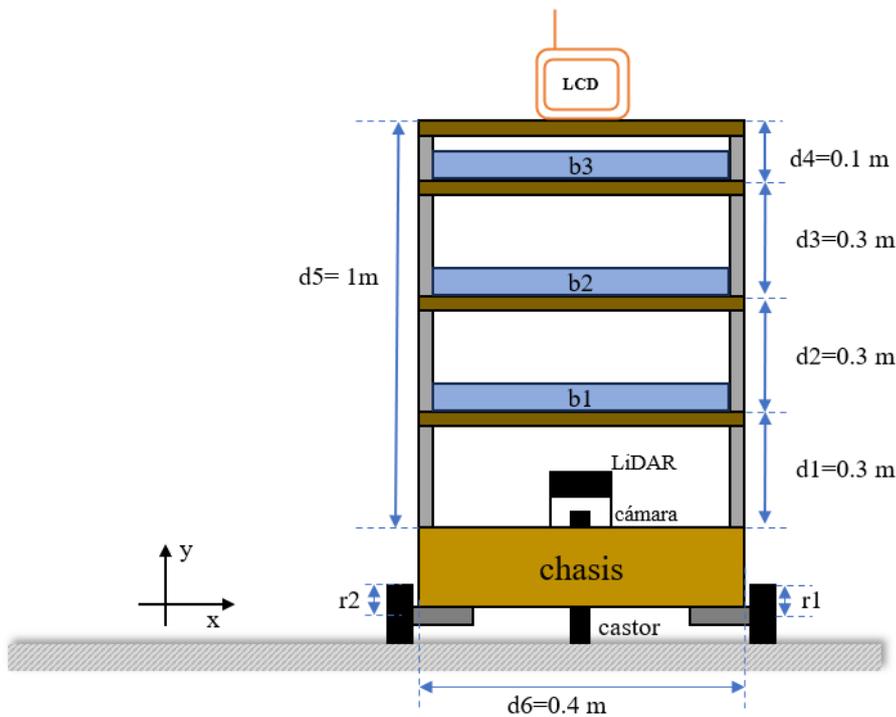
## 2.4 Diseño conceptual

En la **Figura 2.2** se muestra un boceto general de la solución propuesta a la problemática, en donde se observan los componentes principales que forman parte del robot recolector tomando en cuenta los requerimientos de diseño previamente expuestos. Entre los elementos se identifican; un chasis en donde se alojarán la mayor parte de componentes electrónicos, dos ruedas independientes a los lados con sus respectivos motores, una rueda castor en el centro, un soporte

para la cámara, un sensor LiDAR, tres niveles para soportar las bandejas de comida rápida con soportes a los lados para mantener la estructura estable y un soporte en la parte superior para una pantalla LCD.

**Figura 2.2**

Diseño conceptual



Así mismo, se toma en consideración otros parámetros importantes como son la distancia entre cada uno de los soportes para las bandejas, el cual es de 0.3 m entre ellos y de 0.1 m entre el tercer soporte y la base para el soporte de la pantalla LCD.

## 2.5 Parámetros de diseño mecánico

Con la finalidad de forjar un diseño robusto y funcional que responda de manera óptima a las demandas planteadas y asegurar su correcta funcionalidad, se tomaron en cuenta fundamentos teóricos y empíricos en el diseño mecánico. Se realizaron cálculos, simulaciones y análisis que aseguran que el diseño final cumpla con los requisitos establecidos inicialmente.

**Figura 2.3** Vistas generales de la estructura del robot

*Vistas generales de la estructura del robot*



### 2.5.1 Selección de motores

Se utilizaron dos motores DC con escobillas y con *encoders* para el robot. Para la selección se utilizó un catálogo de la familia de motores Pololu. Estos motores incluyen una variedad de tipos, tamaños y características para adaptarse a diversas aplicaciones en robótica, electrónica y automatización. Para la selección de los motores, se tomaron en cuenta principalmente dos parámetros: control de velocidad y torque. En el Apéndice A se fundamenta con mayor detalle teórico y mediante los criterios para esta selección. Las especificaciones de los motores del robot se enlistan en la **Tabla 2.6**.

**Tabla 2.6**

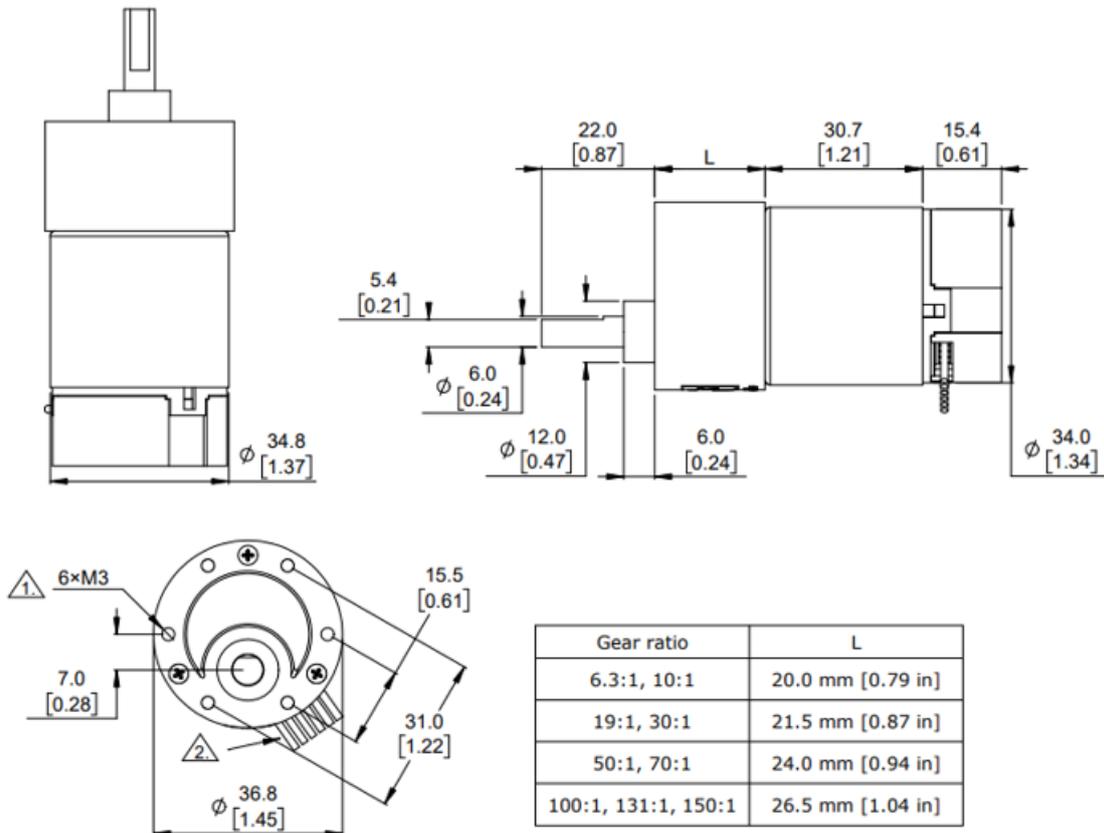
Especificaciones de motores seleccionados

<b>Parámetro</b>	<b>Especificación</b>
<b>Tipo</b>	Motor DC con escobillas
<b>Tensión nominal (V)</b>	12
<b>Corriente de parada (A)</b>	5.5
<b>Corriente sin carga (A)</b>	0.2
<b>Relación de transmisión</b>	131:1
<b>Velocidad sin carga (RPM)</b>	76
<b>Torque de calado (kg-cm)</b>	45
<b>Potencia máxima (W)</b>	6
<b><i>Encoder</i></b>	Efecto Hall de 2 canales

A continuación, en la **Figura 2.4**, se muestran las dimensiones de los motores seleccionados.

**Figura 2.4**

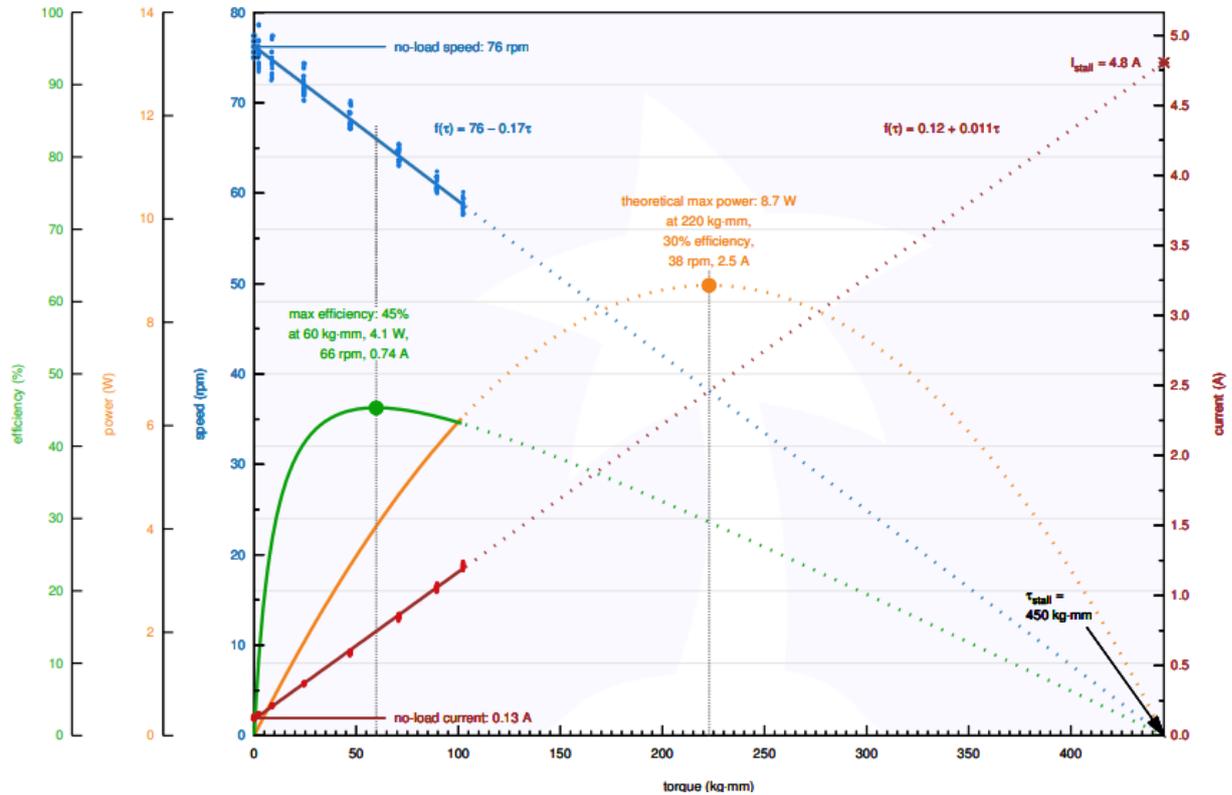
Dimensiones del motor seleccionado



Asimismo, se muestran las curvas de desempeño del motor seleccionado, en donde se observan valores importantes como el torque de calado de 45 kg-cm y velocidad sin carga de 76 RPM.

Figura 2.5

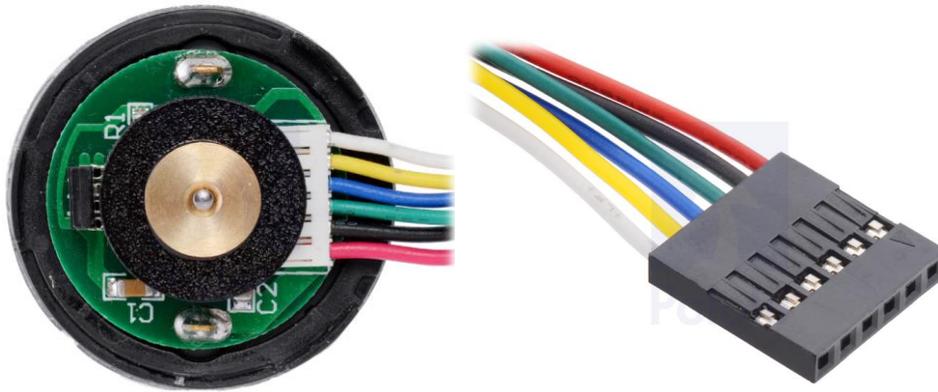
Curvas de desempeño del motor seleccionado a 12V



Adicionalmente, el motor seleccionado emplea un *encoder* de efecto Hall de dos canales para registrar la rotación de un disco magnético que se encuentra en la parte trasera del eje del motor. El *encoder* proporciona datos precisos sobre la posición, velocidad y dirección de los motores, mejorando su capacidad de respuesta. A continuación, se muestra una figura del *encoder* empleado.

**Figura 2.6**

Sensor de efecto Hall de 2 canales



El *encoder* de cuadratura ofrece precisión en el seguimiento del movimiento por medio de una resolución de 64 CPR (conteos/revolución) del eje del motor. Este conjunto incluye seis cables codificados por colores, terminados con un conector hembra. La alimentación del sensor Hall es de 3.5 a 20 V, la cual, junto con las salidas A y B que representan ondas cuadradas desfasadas, permiten determinar tanto la velocidad como la dirección del motor mediante las transiciones de frecuencia y su orden (**Figura 2.7**).

**Figura 2.7**

Salidas A y B del *encoder* del motor seleccionado (12 V)



### 2.5.2 Selección de ruedas y accesorios adicionales

Se seleccionaron ruedas de patines o *scooter* para el robot debido a su versatilidad, durabilidad y adaptabilidad a entornos variables de trabajo. Estas ruedas, al igual que los motores, se encuentran disponibles en el catálogo de Pololu y ofrecen un equilibrio óptimo entre agarre, tracción y resistencia al desgaste. Además, su disponibilidad en el mismo catálogo de Pololu facilitó la integración en el diseño CAD y la compatibilidad con otros componentes del robot. Las especificaciones se muestran en la siguiente tabla.

**Tabla 2.7**

*Especificaciones de ruedas seleccionadas*

<b>Parámetro</b>	<b>Especificación</b>
<b>Diámetro exterior (mm)</b>	144
<b>Ancho (mm)</b>	29
<b>Material</b>	Poliuretano
<b>Peso (g)</b>	235
<b>Color</b>	negro

**Figura 2.8**

*Rueda seleccionada*



## 2.6 Parámetros de diseño electrónico

### 2.6.1 Selección de controlador principal

El controlador que se seleccionó para el correcto funcionamiento del robot recolector fue la Raspberry Pi en su modelo 4B. En este apartado, lo que se buscó fue un controlador de alta potencia de procesamiento capaz de ejecutar sistemas operativos completos, como Linux, y software complejo como ROS.

En el **Apéndice C** se aprecia una tabla comparativa de algunos controladores con sus respectivas especificaciones. De entre todos, el que más destacó fue la Raspberry Pi modelo 4B debido a su capacidad para proporcionar un entorno de computación completo y versátil facilitando el desarrollo, la programación e implementación de sistemas robóticos complejos.

#### Figura 2.9

Raspberry Pi modelo 4B



A continuación, se presentan las especificaciones técnicas del controlador seleccionado.

Tabla 2.8

Especificaciones técnicas del controlador seleccionado (Raspberry Pi modelo 4B)

<b>CPU</b>	<b>1.5GHz 64-bit quad-core Cortex-A72</b>
<b>RAM</b>	1GB, 2GB, 4GB U 8GB LPDDR4
<b>Conectividad</b>	2.4 GHz y 5.0 GHz IEEE 802.11b/g/n/ac WIFI LAN Bluetooth 5.0 BLE
<b>GPIO</b>	40-pin
<b>Puertos USB</b>	2 × USB 3.0 puertos 2 × USB 2.0 puertos
<b>Video y sonido</b>	2 × micro HDMI puertos Puerto de pantalla MIPI DSI de 2 carriles Puerto de cámara MIPI CSI de 2 carriles
<b>Multimedia</b>	1080p60 (H.264)
<b>Potencia de entrada</b>	5V DC vía USB-C 5V DC vía GPIO
<b>Dimensiones</b>	85mm x 53mm

### 2.6.2 Selección de driver para motores

Se seleccionó el *driver* BTS7960 para el motor de 12V y cuya corriente de parada es de 5.5A, tomando en consideración varios factores determinantes. Este controlador fue elegido por su facilidad de uso con Arduino, lo que simplifica su integración en el proyecto. Su capacidad para manejar corrientes considerables asegura un rendimiento adecuado sin alcanzar los límites de su capacidad.

En el Apéndice E, se detallan otras opciones de *drivers* evaluadas, destacando por qué el BTS7960 fue la elección preferida. En la **Tabla 2.9** se presentan las especificaciones de dicho *driver*.

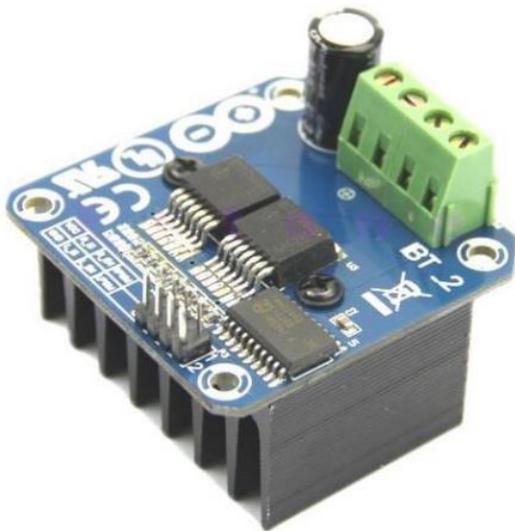
**Tabla 2.9**

Especificaciones del *driver* BTS7960 seleccionado

<b>Especificación</b>	<b>Valor</b>
<b><i>Driver</i></b>	Dual BTS7960
<b>Voltaje de entrada (V DC)</b>	6- 27
<b>Corriente máxima (A)</b>	43
<b>Modo de control</b>	PWM o nivel
<b>Capacidad PWM (kHz)</b>	Hasta 25

**Figura 2.10**

Driver BTS7960 seleccionado



### 2.6.3 Selección de controlador para motores

Para controlar los motores, se utilizó una placa Arduino Nano. Con un total de 22 pines digitales y 8 pines analógicos, el Nano ofrece una cantidad adecuada para el proyecto. Tiene un total de 6 pines PWM disponibles para controlar la velocidad de los motores DC.

### 2.6.4 Selección de cámara y sensor LiDAR

Para implementar un algoritmo de visión utilizando OpenCV, se requiere una cámara para capturar imágenes en tiempo real. Para ello, se seleccionó el módulo de cámara Raspberry Pi 3 como la opción ideal para este propósito. A continuación, las especificaciones.

**Tabla 2.10**

Especificaciones de la cámara seleccionada

Parámetro	Especificación
Dimensiones	25 x 24 x 11.5 mm
Peso	4g
Resolución	11.9 Megapíxeles
Sensor	Sony IMX708
Longitud focal	4.74 mmm
Campo de visión horizontal	66 grados
Campo de visión vertical	41 grados
Área de sensor de imagen	6.45 x 3.63mm

Para la percepción y mapeo del entorno, es fundamental utilizar un sensor LiDAR. Se seleccionó el sensor RPLIDAR A1 para obtener datos precisos de distancia, detección de objetos alrededor del robot y navegación autónoma. En la **Tabla 2.11** se muestran las especificaciones del RPLIDAR A1.

Tabla 2.11

*Especificaciones del sensor LiDAR seleccionado*

<b>Parámetro</b>	<b>Especificación</b>
<b>Rango de distancia</b>	0.15 – 12 m
<b>Rango angular</b>	0 - 360 grados
<b>Resolución de distancia</b>	< 0.5 mm
<b>Resolución angular</b>	≤ 1 grado
<b>Duración de muestreo</b>	0.125 ms
<b>Frecuencia de muestreo</b>	≥ 8000 Hz

### 2.6.5 Consumo de corriente del robot

Se calculó también el consumo de corriente de los componentes del robot. La importancia de este análisis radica en la posterior selección preciso de los elementos de suministro energético, garantizando así un diseño óptimo y eficiente del sistema en su conjunto.

Tabla 2.12

*Consumo de corriente de componentes de 5V*

<b>Componente</b>	<b>Cantidad</b>	<b>Voltaje (V)</b>	<b>Consumo (A)</b>
<b>Raspberry pi</b>	1	5	1.5
<b>Arduino nano</b>	1	5	0.1
<b>Cámara</b>	1	5	0.1
<b>RPLIDAR A1</b>	1	5	1
<b>Driver de motor</b>	2	5	0.1
<b>Pantalla LCD</b>	1	5	1
<b>Total</b>			4.7

### 2.6.6 Selección de regulador de voltaje

Ya que se tienen componentes que operan a 5V, pero los motores funcionan a 12V, fue necesario utilizar un regulador para reducir el voltaje suministrado por la batería a un nivel seguro y compatible para los componentes de 5 V.

Luego de calcular la corriente consumida por los componentes de 5V, se seleccionó un regular capaz de suministrar hasta 5 A aproximadamente. El regulador que se seleccionó es el SKU:DFR0205. Las especificaciones técnicas se muestran en la **Tabla 2.13**

**Tabla 2.13**

Especificaciones del regulador de 5V seleccionado

<b>Parámetro</b>	<b>Especificación</b>
<b>Voltaje de entrada (V)</b>	3.6 - 25
<b>Voltaje de salida (V)</b>	3.3 - 25
<b>Corriente constante de salida (A)</b>	5 V a 5 A
<b>Potencia máxima de salida (W)</b>	25
<b>Frecuencia (KHz)</b>	350
<b>Dimensiones (mm)</b>	46 x 50 x 20 mm

### 2.6.7 Selección de batería

Una vez determinado el consumo eléctrico de los motores y los componentes operativos a 5V, se seleccionó la batería más adecuada para alimentar el robot. Después de un análisis respaldado por los cálculos detallados en el Apéndice E, se decidió utilizar una batería *LiPo* de 3S. Las especificaciones técnicas de esta batería se detallan en la siguiente tabla:

Tabla 2.14

*Especificaciones de la batería seleccionada*

Parámetro	Especificación
Tipo	Polímero de litio
Número de celdas	3
Capacidad (mAh)	7000
Voltaje (V)	11.1
Descarga (C)	100
Dimensiones (mm)	152 x 46 x 28.5
Conector	Enchufe deans

Figura 2.11

*Batería LiPO seleccionada*



## 2.7 Parámetros de diseño computacional

### 2.7.1 Modelo 3D del robot

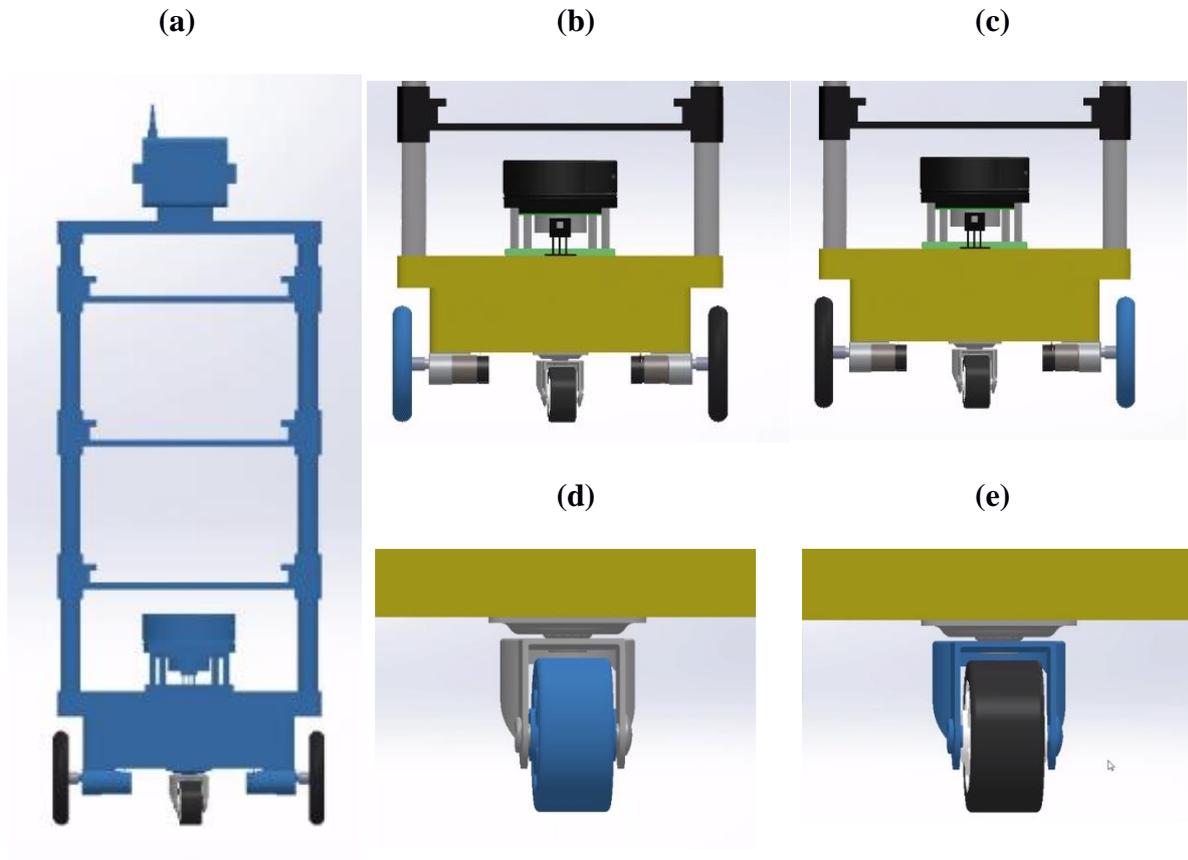
Para describir el modelo del robot en ROS, fue necesario un archivo llamado URDF (*Unified Robot Description Format*). Este archivo describe la estructura, cinemática y geometría de un robot, siendo esencial para su simulación. *SolidWorks 2022* cuenta con una herramienta para exportar el diseño *CAD* del robot, la cual permitió generar el archivo URDF con los eslabones y geometría de este. Como primer paso, la herramienta requirió la definición de la jerarquía de los eslabones (*links*) y articulaciones (*joints*) del robot. Se estableció una estructura jerárquica, en la que el eslabón base llamado por el programa como el "padre", contenía tres "hijos" correspondientes a las tres ruedas del robot.

El eslabón base englobó todo el cuerpo del robot, desde el chasis con sus componentes incluyendo los soportes de las bandejas, culminando en la parte superior con la pantalla. Por otro lado, la rueda castor presentaba dos grados de libertad: un componente torsional debido a su rodamiento y la capacidad de girar sobre su propio eje. Por ello, la rueda castor se dividió en dos eslabones más: la base de la rueda y la rueda en sí misma.

Esta estructura permitió la creación del árbol de transformaciones con la definición precisa de todas las articulaciones y eslabones, proporcionando una representación del robot que fue exportada como modelo *URDF*. En la **Figura 2.12** se puede observar cómo se definieron cada uno de los eslabones del robot.

**Figura 2.12**

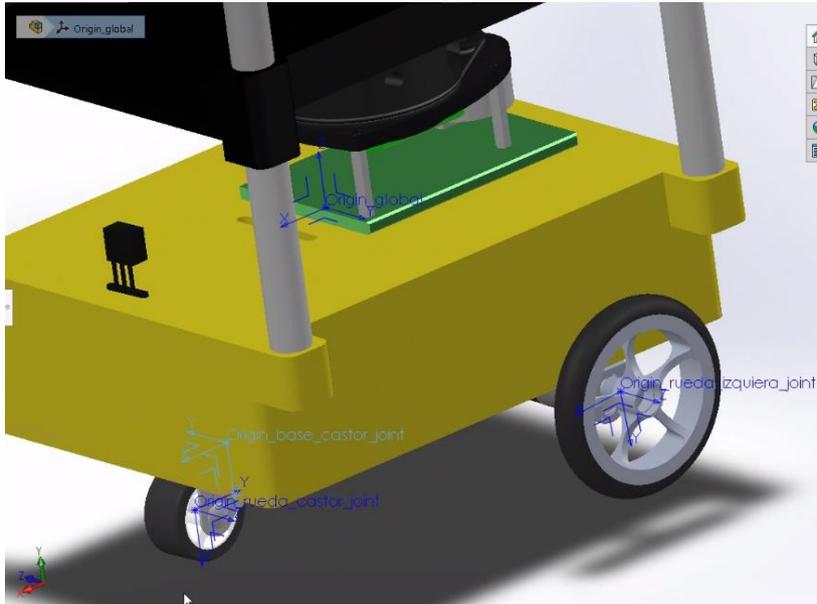
*Sistema de juntas y eslabones del robot: (a) Eslabón base; (b) Eslabón rueda izquierda; (c) Eslabón rueda derecha; (d) Eslabón rueda castor; (e) Eslabón base castor*



También fue necesario establecer los ejes de coordenadas para la correcta definición del robot (**Figura 2.13**). Se definió un eje global centrado en el chasis, siguiendo la convención comúnmente utilizada en la robótica móvil, donde el eje “x” se orienta hacia adelante del robot, el eje “y” representa la dimensión lateral y el eje “z” se dirige verticalmente. En el caso específico de las ruedas, se consideró fundamental que el eje “z” estuviera alineado con el sentido del movimiento de la articulación, siguiendo la regla de la mano derecha. Esta configuración estandarizada de ejes es importante ya que facilita la representación espacial del robot dentro del entorno de simulación.

**Figura 2.13**

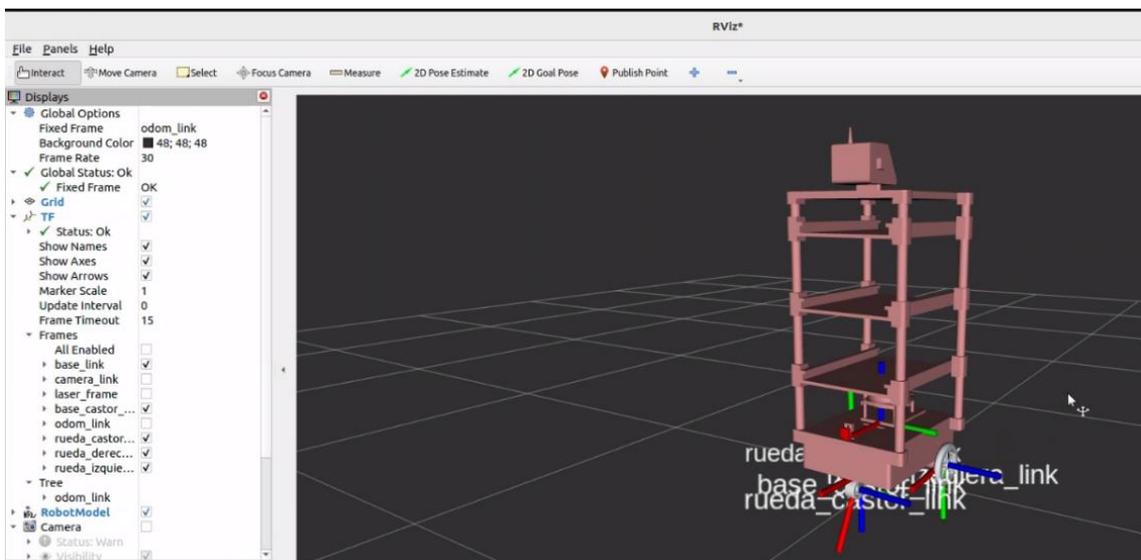
*Sistema de ejes coordenados del robot*



El URDF resultante se proporcionó al nodo *robot\_state\_publisher*, el cual publica los datos en el tema */robot\_description* para acceder a la información sobre la estructura del robot y sus transformaciones (*/TF*). Los valores de las articulaciones se publicaron en el tema */joint\_states*. Se utilizó el *joint\_state\_publisher\_gui*, para verificar el movimiento de las ruedas del robot. En la Figura 2.14 se puede visualizar el robot 3D en Rviz.

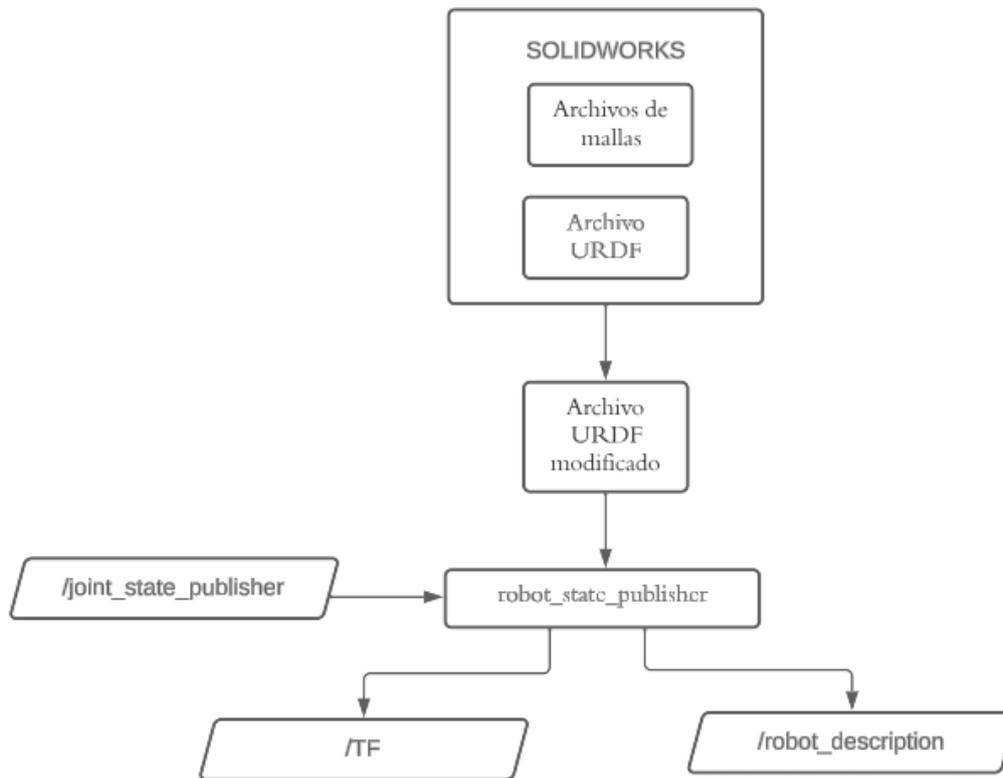
**Figura 2.14**

*Visualización del robot en Rviz*



**Figura 2.15**

*Diagrama de flujo sobre los temas y nodos del modelo 3D del robot*



### 2.7.2 Mapa del entorno de simulación

Se diseñó un entorno del comedor en Gazebo, representando de manera realista el escenario en el cual se moverá el robot. El diseño fue inspirado en el comedor *Fresh Food* ubicado en el campus de la ESPOL.

Figura 2.16

Vista 3D del modelo del comedor utilizado en Gazebo

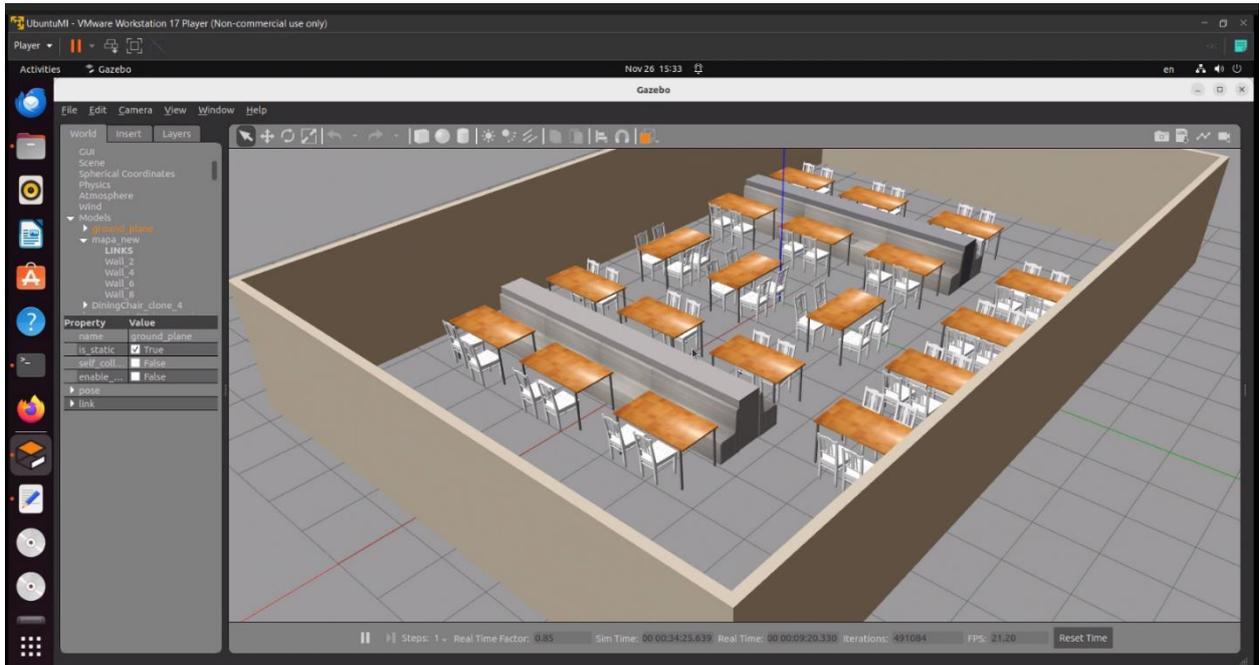
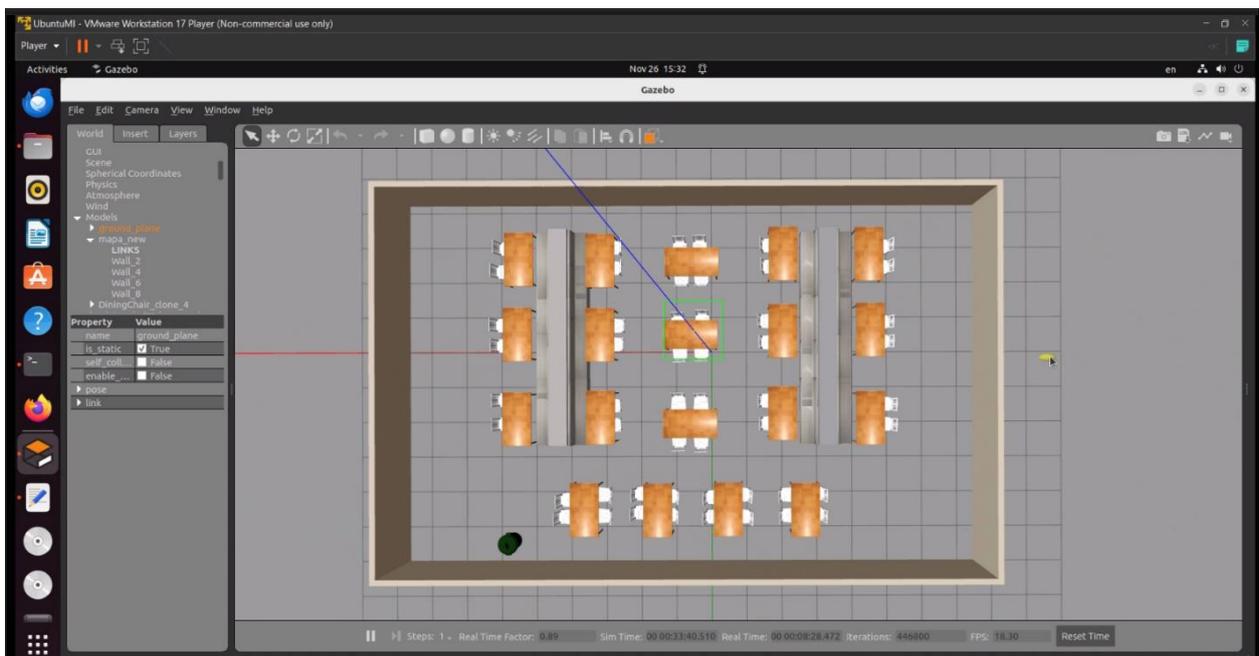


Figura 2.17

Vista superior del modelo del comedor utilizado en Gazebo



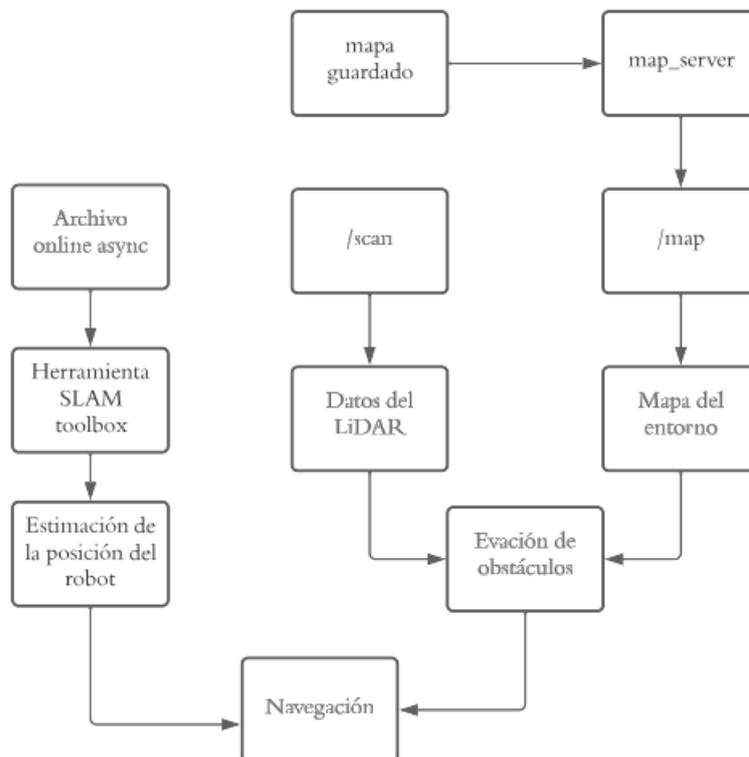
### 2.7.3 Algoritmos de navegación y localización (SLAM)

Para que el robot ejecute una trayectoria de un punto inicial, a un punto final, de manera autónoma, se utilizó el paquete de navegación *Nav2*. Se estableció un tópico llamado *coastmap*, el cual permitió indicar las áreas transitables y obstáculos para que el robot pueda planificar trayectorias seguras.

Para ello, previamente se utilizó un algoritmo de localización y mapeo llamado *SLAM* a través de la librería *slam\_toolbox*, la cual, permitió al robot construir un mapa del comedor, mediante la información receptada por un sensor *LiDAR*. Con un nodo llamado *map\_server* se almacenó el mapa construido y se lo proporcionó para la planificación de rutas y la navegación. Se instaló y utilizó un archivo llamado *online\_async.launch* en donde se configuró todos los nodos que realizaron la construcción del mapa.

**Figura 2.18** Diagrama de flujo para navegación y localización

*Diagrama de flujo para navegación y localización*



#### 2.7.4 Algoritmo de visión en OpenCV

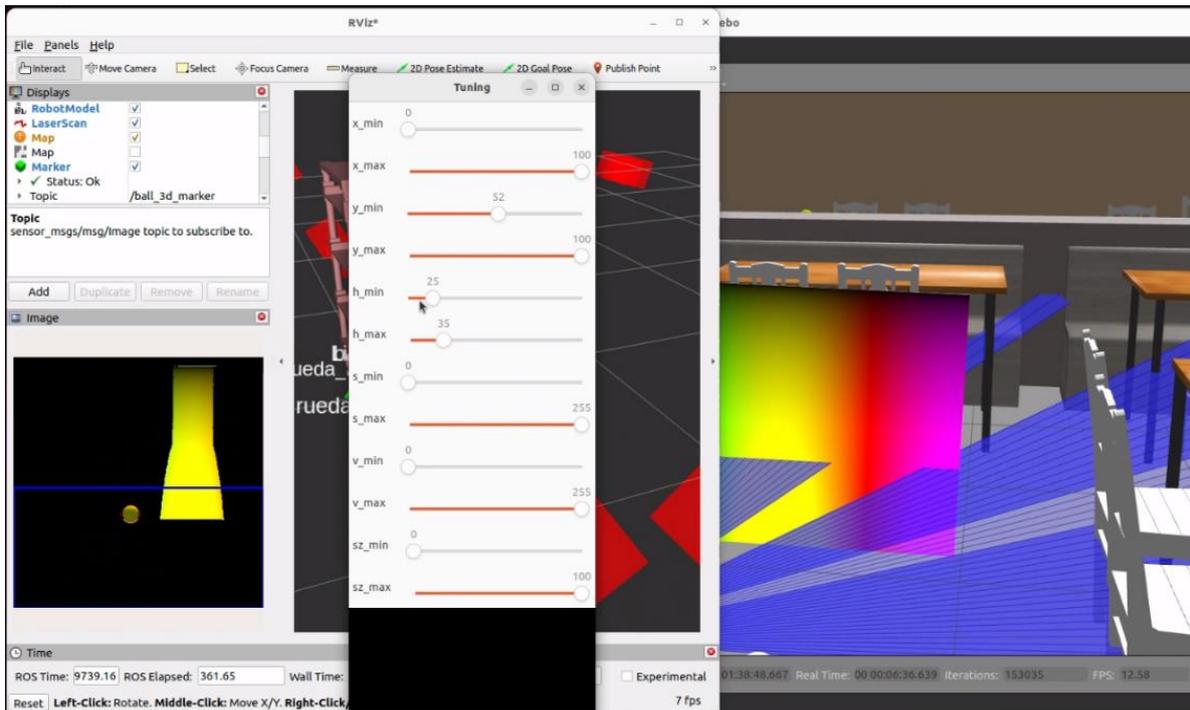
Otra de las funciones autónomas del robot es la detección y seguimiento de objetos, para lo cual se utilizó la biblioteca de código abierto *OpenCV* (*Open Source Computer Vision Library*). Se desarrolló un algoritmo basado en 2 nodos, el primer nodo se encargó del procesamiento de imagen, mediante una cámara y OpenCV, detectando un objeto y devolviendo las coordenadas de este dentro de un marco. El segundo nodo se suscribió a dicha medición, y calculó el comando de velocidad apropiado para enviarlo al sistema de control, es decir a las ruedas.

El objeto que se utilizó fue una esfera de color amarillo, la cual representa el indicador que las personas utilizan, una vez que han terminado de servirse sus alimentos. Para el primer nodo, se procedió a ajustar los parámetros que corresponden a la unidad de procesamiento de imágenes del robot, mediante la activación del *tuning mode=true*. Se ajustaron parámetros HSV (matiz, saturación, brillo), para facilitar la detección del indicador basado en su color. Para ello, se agregó en *Rviz* un tema llamado *image\_topic* para especificar la entrada de imágenes, y se seleccionó *image\_tuning*.

Durante el ajuste de parámetros, el robot con su cámara fue puesto frente al objeto y los parámetros fueron ajustados de manera iterativa para optimizar la detección del objeto en la imagen. Se introdujo en Gazebo un componente denominado *spectrum* con el objetivo de definir la detección del objeto en un color específico.

**Figura 2.19**

*Ajuste de parámetros HSV para detección de objeto*



En la Figura **Figura 2.19** se observa el proceso de ajuste de parámetros HSV y marco de visión de la cámara ( $x_{min}$ ,  $x_{max}$ ,  $y_{min}$ ,  $y_{max}$ ) a través de la ventana *Tuning*. En la parte izquierda de la imagen se muestra el tema (*topic*) de la cámara en donde se observa el indicador (esfera color amarillo) siendo detectada por la cámara y de fondo el espectro de colores siendo ajustado al color del objeto. Los parámetros de ajuste se registraron en un archivo YAML designado.

### 2.7.5 Distribución general de transformadas del robot

Finalmente, a través del árbol de transformadas obtenido mediante un plugin de ROS llamado *rqt\_tree*, y mostrado en las **Figura 2.20** y **2.21**, se muestra una representación gráfica de las relaciones de transformación entre los distintos componentes del robot.

Figura 2.20

Árbol de transformadas del robot Parte - 1

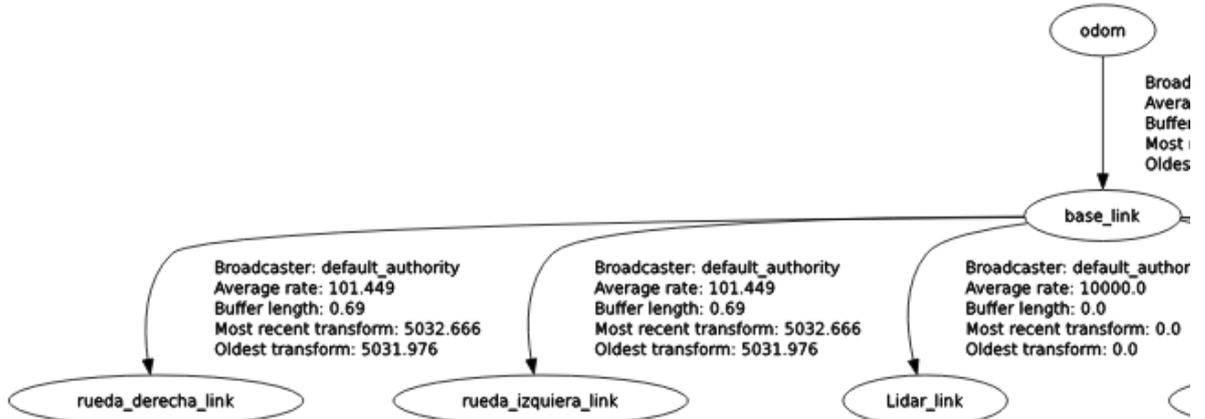
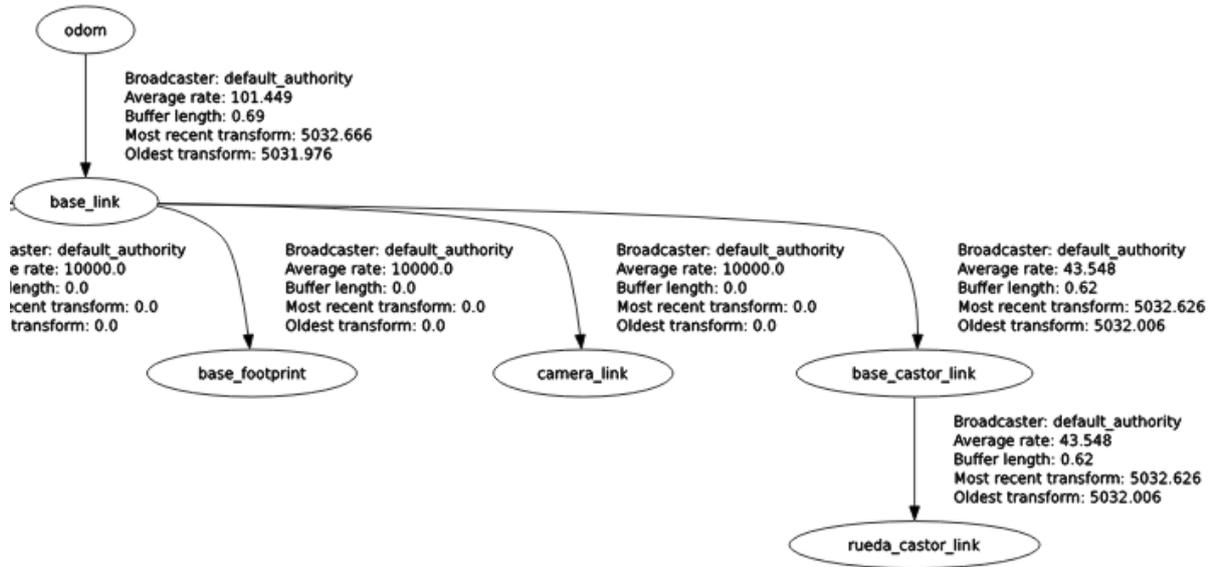


Figura 2.21

Árbol de transformadas del robot - Parte 2



## **Capítulo 3**

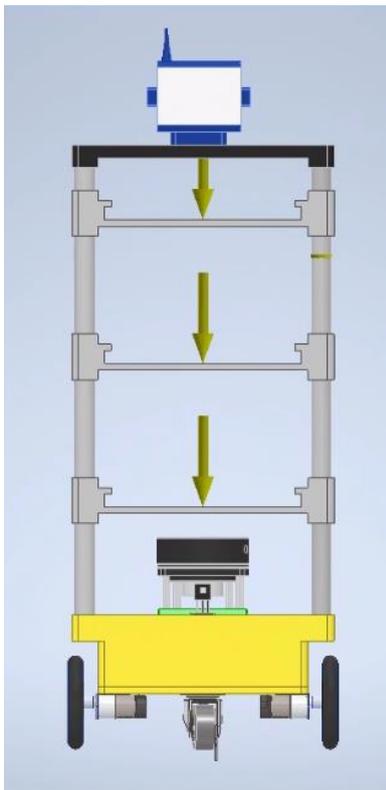
### 3. RESULTADOS Y ANÁLISIS

En este capítulo se presentan los resultados y análisis de la solución propuesta, abordando tres aspectos fundamentales: diseño mecánico, eléctrico y computacional. Asimismo, se incorpora un análisis de costos evaluando el impacto económico del proyecto.

#### 3.1 Diseño mecánico

**Figura 3.1**

*Diseño 3D del robot recolector con cargas aplicadas*



En la **Figura 3.1** se muestra la vista frontal del diseño 3D del robot recolector planteado. Además, se incluyeron tres cargas, cada una de 3 kg, las cuales representan la fuerza que ejercen las bandejas de comida rápida. Según [28] una comida de almuerzo, con todos sus platos incluidos, tales como; plato fuerte, sopa, ensalada, bebida y postre, pesa alrededor de 1.6 kg. En la siguiente

tabla se muestra un desglose de los principales componentes que ingiere una persona normal en su dieta.

**Tabla 3.1**

*Peso de alimentos en una dieta de una persona natural*

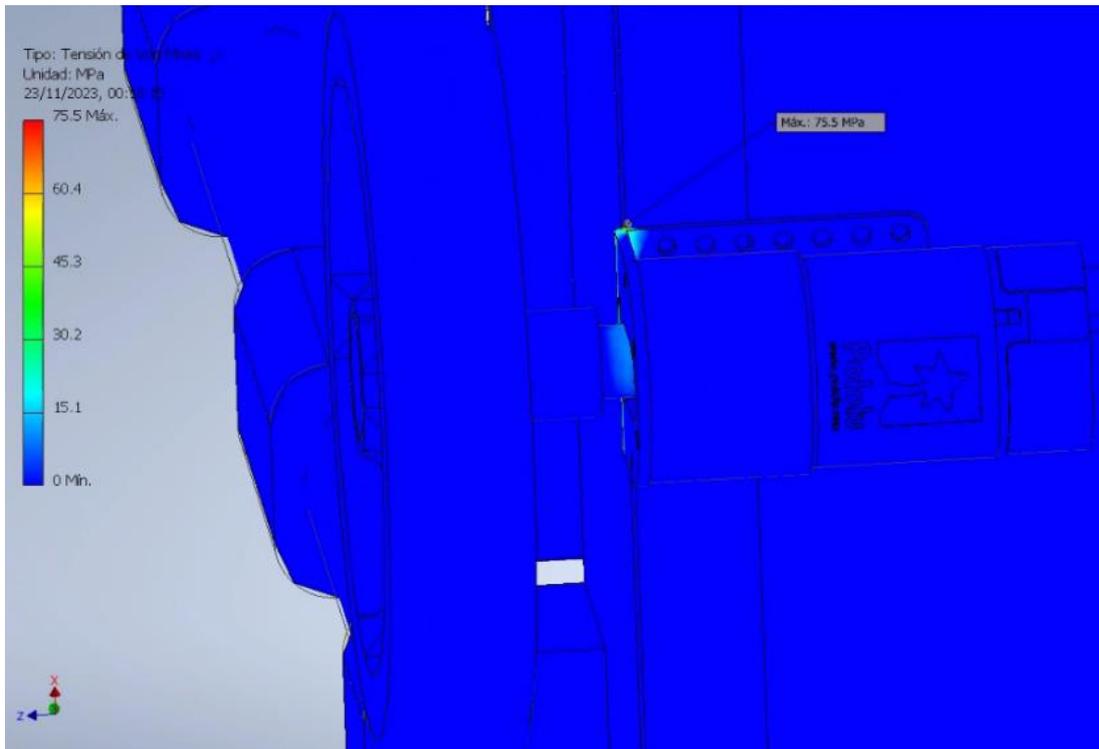
<b>COMPONENTE</b>	<b>PESO (KG)</b>
<b>BEBIDA</b>	0.5
<b>Frutas y postre</b>	0.15
<b>Pan</b>	0.12
<b>Plato principal</b>	0.5
<b>Sopa</b>	0.15
<b>Ensalada</b>	0.15
<b>Total</b>	1.57

A partir de este valor, se estableció que cada bandeja será capaz de soportar un peso máximo de 3 kg, es decir, casi el doble del peso de una dieta de un almuerzo de una persona natural.

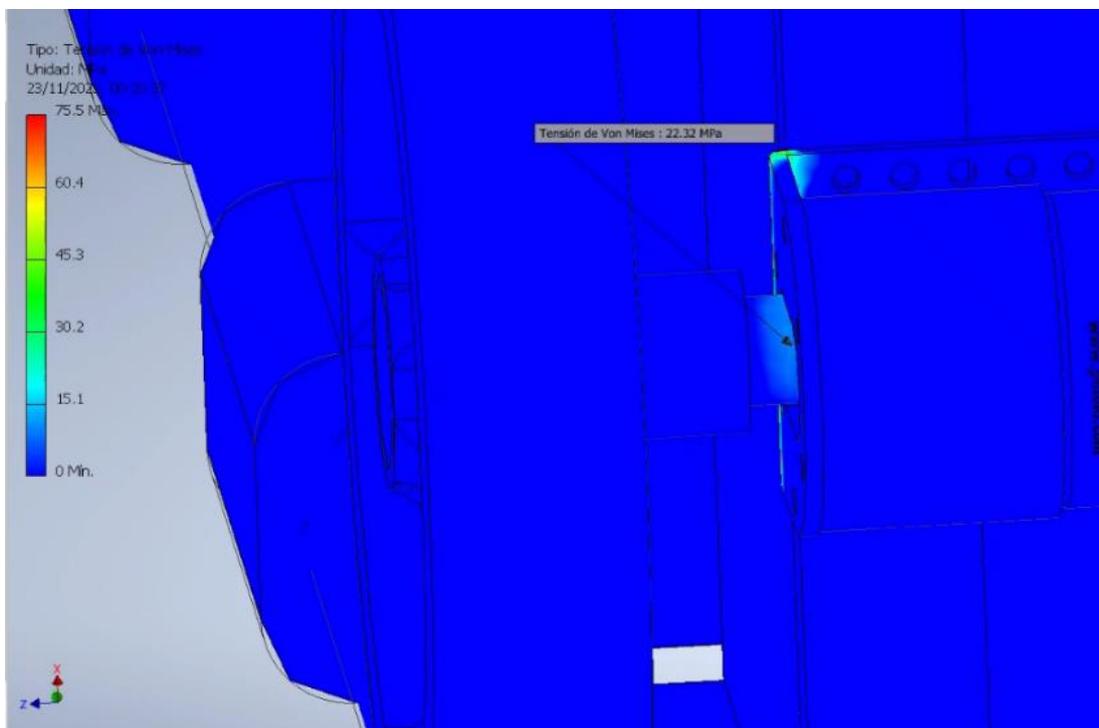
Luego, se realizó un análisis de elementos finitos (FEA), para determinar las cargas soportadas por el robot.

**Figura 3.2**

*Análisis de tensión de Von Mises - Soporte de motor*

**Figura 3.3**

*Análisis de tensión de Von Mises - eje de motor*



En la **Figura 3.2** y **Figura 3.3** se observa el esfuerzo de Von Mises sobre el soporte del motor y eje del motor respectivamente. Estos componentes mecánicos son los que mostraron los valores de esfuerzo más altos y por lo tanto el análisis indica que son los elementos que soportan una mayor carga. En el soporte del motor el esfuerzo de Von Mises registrado fue de 7.5 MPa mientras que en el eje del motor se tuvo un esfuerzo de 22.32 MPa.

**Tabla 3.2**

*Propiedades mecánicas del aluminio*

### Propiedades

<b>Material</b>	<b>Aluminio</b>
<b>Resistencia a la tensión (<math>S_{ut}</math>)</b>	<b>310 MPa</b>
<b>Resistencia a la fluencia (<math>S_y</math>)</b>	<b>276 MPa</b>

$$FS = \frac{\sigma_y}{\sigma_{Von\ Mises}} \quad (3.1.1)$$

$$FS = \frac{310}{7.5} = 4$$

En la **Tabla 3.2** se muestran las propiedades mecánicas del aluminio, material del cual está elaborado el soporte del motor, según el fabricante Pololu. El factor de seguridad calculado para este componente fue de 4, el cual, al ser mayor a 1, indica que este elemento es lo suficientemente resistente y seguro bajo las condiciones de carga previstas.

**Tabla 3.3**

*Propiedades mecánicas del acero*

### Propiedades

<b>Material</b>	<b>Acero</b>
<b>Resistencia a la tensión (<math>S_{ut}</math>)</b>	<b>420 MPa</b>
<b>Resistencia a la fluencia (<math>S_y</math>)</b>	<b>350 MPa</b>

$$FS = \frac{\sigma_y}{\sigma_{Von\ Mises}} \quad (3.1.2)$$

$$FS = \frac{420}{22.32} = 19$$

De igual forma, se realizó el cálculo de factor de seguridad para el eje del motor, cuyo esfuerzo de Von Mises registrado, fue de 22.32. En este caso, el material del cual está hecho el eje del motor de marca Pololu, es de acero, cuya resistencia a la fluencia es de 350 MPa. Así, se obtuvo un FS de 19, del cual se puede concluir que dicho componente es capaz de resistir las cargas aplicadas, por lo que se considera seguro al diseño.

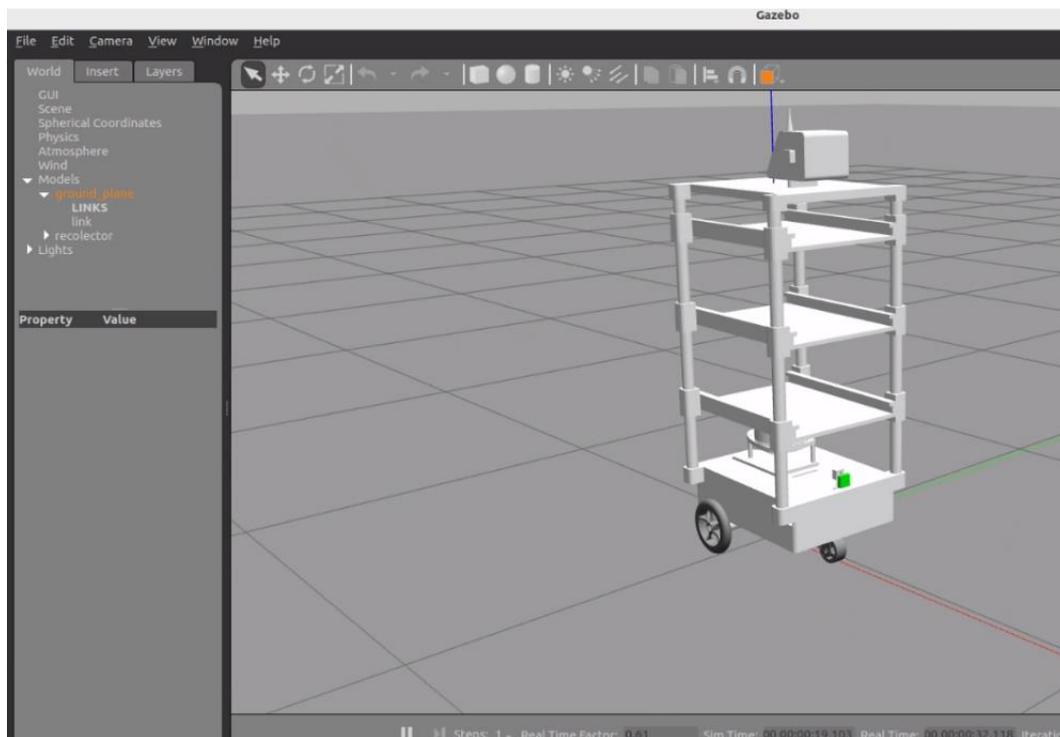
Se tomó al *PLA* como el material para el diseño mecánico del robot debido a sus cualidades en la impresión 3D dentro de las cuales destacan su dureza, durabilidad y facilidad de impresión. Además, su capacidad para resistir impactos y flexiones, lo convierten en una opción óptima para los componentes estructurales en diversas aplicaciones robóticas, satisfaciendo las exigencias mecánicas del proyecto.

### 3.2 Diseño computacional

Para poder visualizar el robot en Gazebo (**Figura 3.4**) fue necesario definir algunas propiedades físicas del robot como la masa y la inercia. Para el control del robot se utilizó un plugin llamado *diff\_drive*, el cual se utiliza para simular un sistema de tracción diferencial en un robot móvil. Se declararon las respectivas articulaciones de las ruedas, distancia y diámetro de las ruedas, aceleración, y otro tópico para la odometría del robot. El eslabón *odom\_link*, permite conocer información estimada sobre la posición y orientación del robot en función de la odometría, esto sería, en la vida real, el equivalente a los datos proporcionados por los *encoders* de los motores de las ruedas.

**Figura 3.4**

*Visualización del robot en Gazebo*



En cuanto a la navegación autónoma del robot, el primer paso fue realizar un mapeo del entorno del comedor. Para ello se utilizó la herramienta de *slam\_toolbox* junto con el archivo de *mapper\_params\_online\_async.yaml* el cual es el algoritmo por excelencia de mapeo en entornos

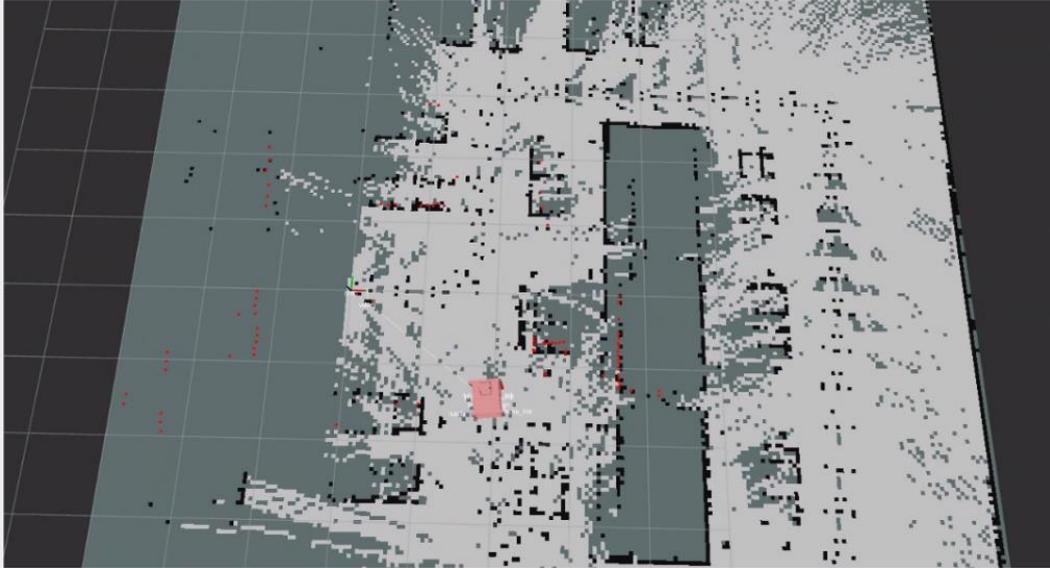
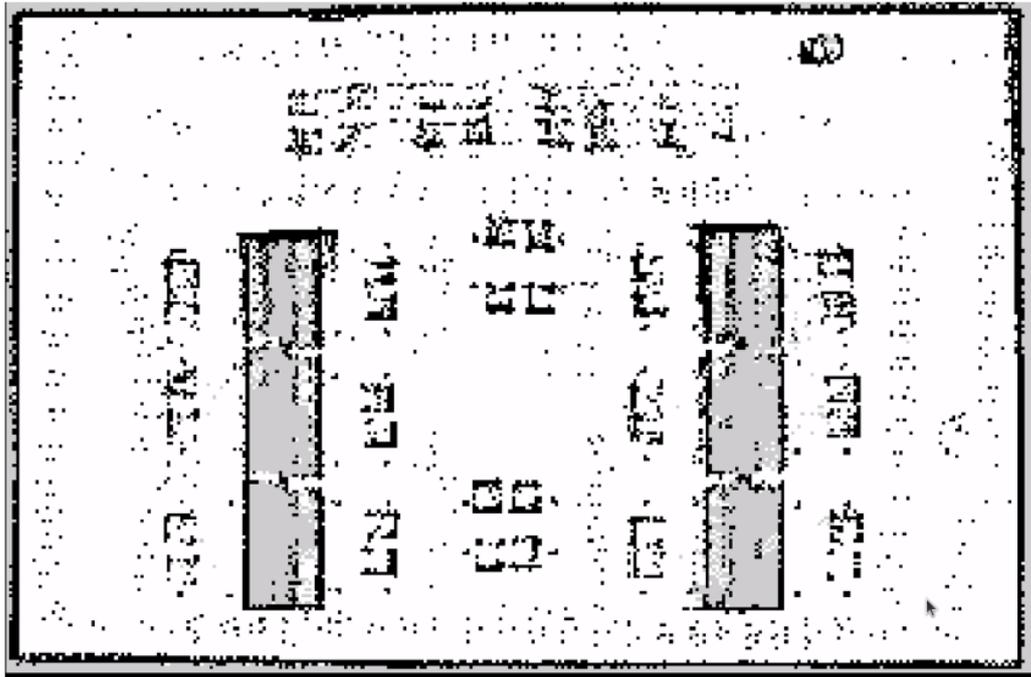
robóticos. El término “*online*” permitió la adaptación dinámica de los parámetros del algoritmo de mapeo en tiempo real, tales como: marco de odometría, marco del mapa, tópico del sensor *LiDAR*, y el modo. El término “*async*” garantizó que se procese siempre el escaneo más reciente del sensor para evitar retrasos. Luego de ejecutar el nodo *SLAM toolbox*, se recopilaron los datos del sensor *LiDAR* moviendo el robot por todo el comedor. Para mover el robot se utilizó una herramienta llamada *rqt\_steering*, la cual se utiliza para controlar la velocidad lineal y angular de un robot móvil mediante una interfaz visual e intuitiva. a través del envío de mensajes en el tema */cmd\_vel*, publicado en el plugin del robot diferencial.

En la **Figura 3.5 (a)** se muestra el proceso de recorrido del robot para la recopilación de datos del sensor mientras se crea a la par un mapa en tiempo real.

Finalmente, en la **Figura 3.5 (b)** se muestra el mapa generado el cual se guardó como un archivo *yaml*.

**Figura 3.5**

*Mapa del entorno realizado con SLAM: (a) Proceso de mapeo; (b) Resultado*

**(a)****(b)**

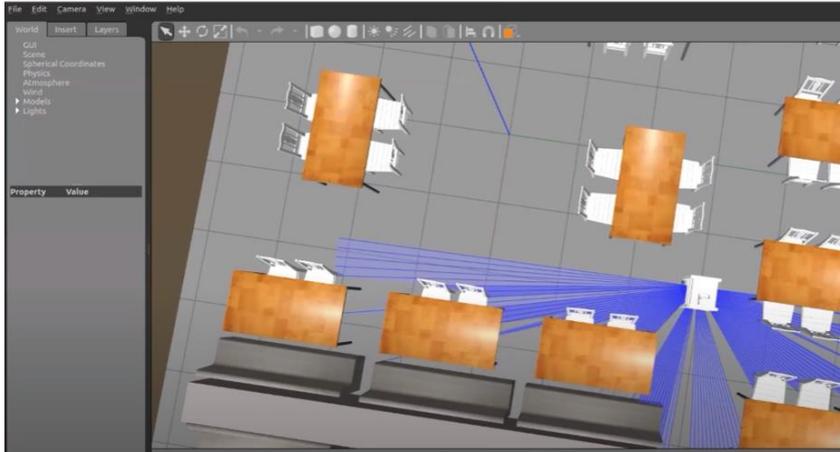
Una vez generado el mapa se utilizó un algoritmo de localización llamado ACML (*Adaptive Monte Carlo Localization*). Esta herramienta realiza una estimación de la posición de un robot en un entorno previamente conocido, lo que permitió la localización y actualización de la posición del robot mientras se mueve en el comedor. Este algoritmo recibe el mapa previamente guardado, y lo publica en el tema */map*, visto desde *Rviz*. Luego se corre el algoritmo *amcl* y se envía una pose estimada desde *Rviz* en la dirección en donde se encuentra el robot en el comedor.

En la **Figura 3.6** se observa el resultado del algoritmo de localización. En la parte **(a)** se muestra una ubicación aleatoria del robot en el comedor, luego, en *Rviz*, se le envió una pose estimada con la herramienta *2D Pose Estimate* **(b)**, finalmente, se validó que el algoritmo estimó instantáneamente la posición del robot, localizándolo con relación al mapa conocido **(c)**.

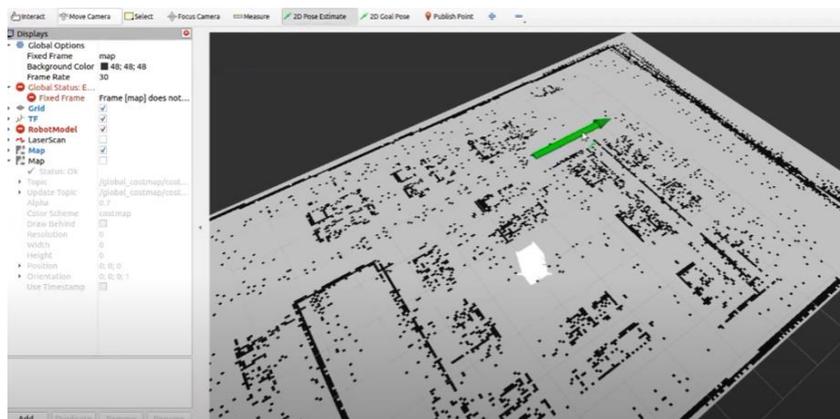
Asimismo, el resultado obtenido del desarrollo del algoritmo de navegación autónoma se muestra en la **Figura 3.7**. De igual forma que en localización, se reusó el mapa generado, esta vez con un parámetro adicional: *map\_subscribe\_transient\_local:=true*, con lo cual se inicializa el paquete *nav2* y se habilita la suscripción al mapa de forma local y transiente. En la parte **(a)** de la figura, se muestra al robot ubicado en el pasillo de en medio del comedor y, usando el comando *2D Goal Pose* **(b)**, se le indicó una ubicación y orientación deseada en el mapa. Finalmente, el robot navegó a dicho punto encontrando la ruta óptima **(c)**.

**Figura 3.6**

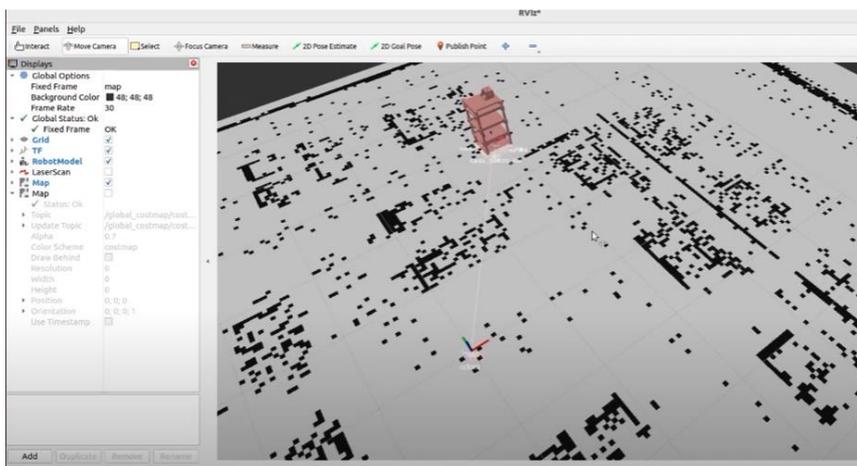
Algoritmo de localización: (a) Posición inicial del robot en Gazebo; (b) Envío de pose estimada; (c) Localización de robot en pose estimada



(a)



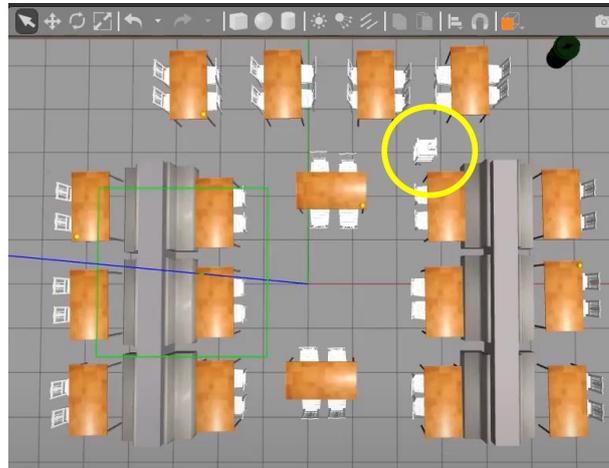
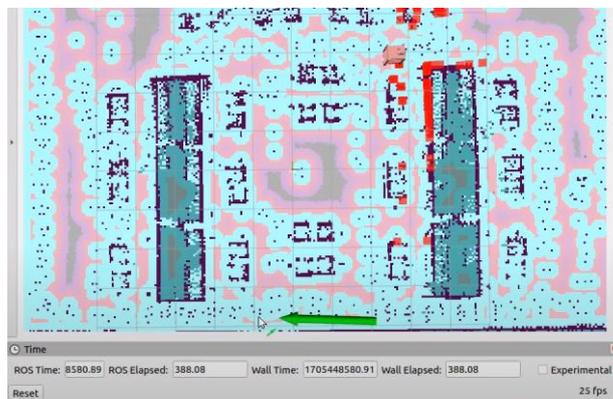
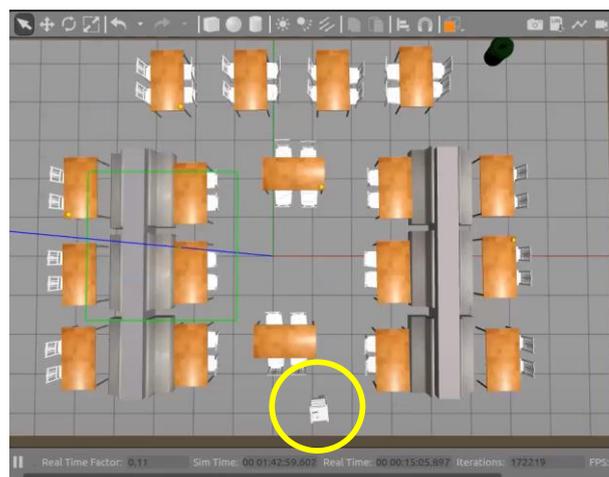
(b)



(c)

**Figura 3.7**

*Algoritmo de navegación: (a) Posición inicial del robot en Gazebo; (b) Envío de objetivo estimado; (c) Posición final del robot*

**(a)****(b)****(c)**

Respecto al algoritmo de detección y seguimiento de objetos con OpenCV, luego de obtener las coordenadas del objeto indicador mediante el primer nodo como se mencionó previamente en el apartado 2.7.4, se utilizó un segundo programa para el seguimiento del robot hacia la esfera. El nodo *follow\_indicator* se encargó de que el robot siga a la esfera detectada en el entorno virtual. Mediante mensajes del tópico */detected\_indicator*, se actualizó la posición de la esfera en el marco de la cámara, para ajustar la velocidad del robot y seguirla, publicando comandos de velocidad en */cmd\_vel*.

En resumen, el robot ajusta dinámicamente su velocidad y dirección para seguir el objeto indicador detectado o activar una estrategia de búsqueda este no está presente. Funciona con las siguientes condiciones:

**Cuando se detecta el indicador:**

- Si el objeto está cerca al robot ( $self.target\_dist < self.max\_size\_thresh$ ), entonces el robot ajusta su velocidad lineal (*linear.x*) y se mueve hacia adelante para seguir el objeto.
- La velocidad angular (*angular.z*) se calcula en función de la posición horizontal del objeto (*target\_val*) y un multiplicador angular (*angular\_chase\_multiplier*).

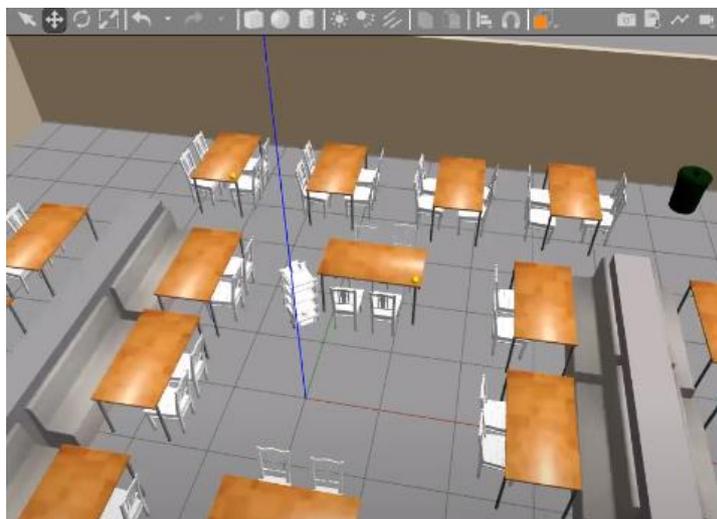
**Cuando no se ha detectado el indicador recientemente:**

- El robot activa una estrategia de búsqueda rotacional estableciendo una velocidad angular fija (*search\_angular\_speed*).

En la **Figura 3.8** se muestra el resultado cuando el robot con su cámara detecta el objeto indicador **(a)** y **(b)** en su posición inicial de su recorrido **(c)**. Posteriormente, en la **Figura 3.9** se aprecia la posición final de robot una vez que se ha acercado al objeto **(b)** y lo que observa su cámara **(a)**.

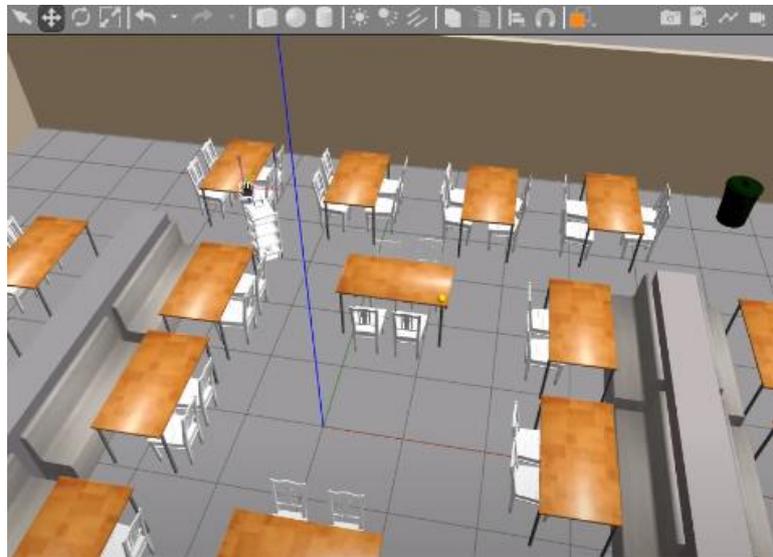
**Figura 3.8**

*Algoritmo de OpenCV, P1: (a) y (b) Detección de objeto; (c) Posición inicial del robot*

**(a)****(b)****(c)**

**Figura 3.9**

*Algoritmo de OpenCV, P2: (a) Detección de objeto; (b) Posición final del robot*

**(a)****(b)**

### 3.3 Análisis de costos

Es importante también conocer cuál es la inversión inicial que se requiere para el proyecto. En la **Tabla 3.4** se desglosa los costos de cada uno de los elementos que forman parte del proyecto el cual está dividido en tres grandes tipos: mecánico, eléctrico y computacional. Este ofrece una visión detallada de los gastos directos e indirectos, y permite identificar las áreas donde se pueden reducir costos, mejorar la eficiencia o reasignar recursos para optimizar el proyecto.

El mayor rubro se encuentra en los componentes computacionales ya que se requiere de programadores especializados en algoritmos de localización, navegación, y visión por computadora. El presupuesto asignado está en función del número de días en que se espera cumplir con cada uno de los algoritmos. Así, un programador desarrollador senior de ROS, gana en promedio \$1000 mensuales, sin embargo, en un contrato por proyecto, se espera que en 10 días laborales, desarrolle la solución deseada. El rubro del componente mecánico está en segundo lugar principalmente por el número de horas que demanda la impresión 3D de todos los elementos que forman parte de la estructura del robot.

Tabla 3.4

Tabla de costos de los componentes del robot

Tipo de componente	Descripción	Cantidad	Precio Unitario	Precio Total
<b>Mecánico</b>	Impresión 3D	268 h	1.80	482.4
	Rollo de filamento PLA	9	23	207
	Rueda de marca Pololu de tipo Scooter 144×29mm - color negro	2	7.95	15.9
	Rueda castor	1	45	45
	Pernos y accesorios adicionales	N/A	9.95	9.95
	Motor de marca Pololu de tipo Metal Gearmotor 37Dx73L mm 12V con encoder tipo Hall	2	51.95	103.9
	<b>Eléctrico y electrónico</b>	Microcontrolador Raspberri Pi modelo 4B	1	78
Batería 3S <i>LiPo</i> - 7000mAh		1	55.99	55.99
Regulador de 5V		1	8.5	8.5
Switch		1	3.5	3.5
Pack de jumpers macho y hembra		2	5.99	5.99
Sensor LiDAR RPLiDAR A1		1	99	99
Driver BTS7960		2	13.88	27.76
Arduino Nano		1	9.2	9.2
Módulo de cámara Raspberri Pi 3		1	29.95	29.95
Pantalla LCD		1	43.99	43.99
<b>Computacional</b>	Dibujante de software CAD	1	100	100
	Programador de entorno en gazebo	1	500	500
	Programador de algoritmo de navegación autónoma	1	600	600

	Programador de algoritmo de visión artificial	1	600	600
<b>Gastos extras</b>	Instalación y puesta en marcha	1	300	300
	Mantenimiento	1	100	100
			<b>Total</b>	<b>3426.03</b>

## **Capítulo 4**

## 4. Conclusiones y recomendaciones

### 4.1 Conclusiones

- En el componente mecánico, se diseñó una estructura mecánica adaptativa en forma de soporte para facilitar la recolección de bandejas de comida rápida en comedores cuyas medidas generales es de 110 cm x 40 cm x 42 cm. La capacidad inicial de bandejas que se consideró para el diseño del robot es de 3 bandejas estándar con medidas de 42 cm x 32 cm, capaces de soportar hasta 3kg cada una de ellas. Se contempló una separación de 21 cm aproximadamente entre cada uno de los niveles. El material con el que diseñaron los soportes, así como los demás componentes mecánicos que forman parte de la estructura del robot, desde el chasis hasta la parte superior incluyendo una base para una pantalla *LCD*, fue el *PLA*, con el cual se obtuvo un peso máximo de 28 kg. Así, luego de realizar los cálculos respectivos de cinemática, y dinámica, se seleccionaron los motores con un torque de parada de 45 kg.cm para unas ruedas de *scooter* de 144 cm de diámetro. Finalmente, se realizó un análisis estático, de donde se pudo garantizar que la estructura y sus componentes son seguros y capaces de soportar las cargas a las que se enfrentarán en la operación del robot.
- Con respecto al diseño electrónico, se logró seleccionar una fuente de alimentación para garantizar el rendimiento óptimo de los motores y componentes electrónicos del robot. Con una capacidad de 7000 mAh, esta batería *LiPO* cumplió con la capacidad de carga del robot, la cual es de 13.5 A en capacidad máxima de operación, y de 4.56 A en condiciones normales de operación. Dichos valores se calcularon considerando la carga de los componentes que trabajan a 5 V y la corriente de parada de los motores, de 5.5 A según las especificaciones del fabricante para condiciones máximas, y de 2.28 A en condiciones normales. Así, la batería permitió una autonomía de 31.1 min en condiciones máximas, y 1.53 h en condiciones normales, lo cual cumple con éxito el objetivo planteado.

- En cuanto al diseño computacional, se desarrolló un modelo de mapeo y navegación autónoma para el robot móvil. Para el primero, se replicó el diseño de un comedor localizado en la ESPOL, el cual consta de 2 pasillos laterales con 6 mesas en cada uno, un pasillo central con 3 mesas, y un pasillo en la parte trasera, con 4 mesas. Para el diseño del ambiente virtual se utilizó Gazebo. El sistema de navegación autónoma se implementó a través de un algoritmo *SLAM* y localización *AMCL*, empleando un sensor *LiDAR* para la percepción del entorno. Inicialmente, el robot se desplazó por el mapa mediante un plugin llamado *rqt\_steering*, generando datos de odometría y mapeo. A la par, el algoritmo *SLAM* se encargó de construir un mapa detallado del entorno, y finalmente, el algoritmo *AMCL* posibilitó la localización precisa en tiempo real del robot en dicho mapa. Esta capacidad permitió al robot navegar de forma autónoma y evitar obstáculos gracias a la información proporcionada por el *LiDAR*. Para la validación, se enviaron posiciones estimadas al robot, y éste logró cumplirlas con bastante precisión.
- Se implementó en un ambiente virtual un sistema de reconocimiento e identificación de objetos para localizar mesas con bandejas usadas utilizando una cámara y la plataforma OpenCV. Para esta tarea se utilizaron dos programas principales. El primero, se encargó de la detección de imágenes, devolviendo las coordenadas del objeto indicador. Se utilizó una esfera en Gazebo de 3 cm de radio color amarilla, y, mediante un tópicio llamado */image\_tuning* se ajustaron los parámetros correspondientes a los límites observables (en x y y) de la cámara y HSV. El segundo nodo o programa, manejó el cálculo y envió del comando de velocidad al control del robot, en el tópicio */cmd\_vel*, con lo cual el robot, considerando la información sobre la posición del indicador, ajustó su velocidad angular y lineal para controlar su movimiento. Para verificar el objetivo de este algoritmo, se colocaron algunas esferas como indicador en ciertas mesas del comedor en Gazebo, y el robot se dirigió a cada una de ellas, validando así el funcionamiento autónomo del robot recolector de bandejas de comida usadas.

## 4.2 Recomendaciones

- Para el diseño mecánico, se recomienda explorar materiales de impresión 3D alternativos, tales como el ABS debido a su mayor resistencia y durabilidad. Además, algunas piezas mecánicas diseñadas en este trabajo, tales como: la base del chasis, la parte superior del chasis, y el soporte para las bandejas, resultan ser componentes únicos, por lo que el tiempo de impresión es muy alto y podría afectar en el desgaste de la impresora. Así, para optimizar este proceso, se recomienda realizar una fragmentación por partes, lo cual no sólo reducirá el tiempo de impresión de los componentes, sino también minimizará posibles inconvenientes asociados al proceso extenso de impresión.
- En cuanto al algoritmo de detección y seguimiento de imágenes en *OpenCV*, se recomienda utilizar otro tipo de indicador para las mesas, tales como un objeto rectangular o cilíndrico para validar si el algoritmo propuesto funciona de manera eficiente o sería necesario agregar nuevos parámetros y/o funciones dependiendo de la geometría del objeto. Asimismo, resultaría interesante realizar una técnica de seguimiento avanzado basado en métodos numéricos como la trigonometría, para calcular la ubicación del objeto en tiempo real mediante el tamaño del objeto y las características de la cámara utilizada.
- Se han abordado temas de autonomía tales como navegación y visión por computadora, sin embargo, se sabe que las máquinas no son 100% eficientes todo el tiempo, por tanto, controlar al robot de forma manual puede ser requerido, sobre todo cuando se quiere realizar una prueba, validación o configuración. Así, se recomienda implementar un sistema de tele operación para controlar al robot de forma manual, ya sea con un mando de videojuegos, consola, joystick, etc. Como apoyo, ROS cuenta con un paquete llamado *joy*, el cual facilita la integración de dispositivos de entrada como los antes mencionados, con el sistema de ROS.

## Referencias

- [1] R. Barua and S. Datta, “Modernization of Robotics Application in 21st Century: A Review,” *Journal of Mechanical Robotics*, vol. 5, no. 2, pp. 24–34, Sep. 2020, doi: 10.46610/jmmdm.2020.v05i02.005.
- [2] C. Cheng, J. Fu, H. Su, and L. Ren, “Recent Advancements in Agriculture Robots: Benefits and Challenges,” *Machines*, vol. 11, no. 1. MDPI, Jan. 01, 2023. doi: 10.3390/machines11010048.
- [3] J. Bernal, “Robots industriales: revolucionando la industria moderna.” Accessed: Nov. 12, 2023. [Online]. Available: <https://www.metalmecanica.com/es/noticias/robots-industriales-revolucionando-la-industria-moderna>
- [4] M. Muro, R. Maxim, and Whiton Jacob, “Automation and Artificial Intelligence: How machines are affecting people and places,” *Brooking*, Jan. 2019.
- [5] R. Alt, “Digital Transformation in the Restaurant Industry: Current Developments and Implications,” *Journal of Smart Tourism*, vol. 1, no. 1, pp. 69–74, Mar. 2021, doi: 10.52255/smarttourism.2021.1.1.9.
- [6] Y. Fernando, A. Mathath, and M. A. Murshid, “Improving Productivity: : A Review of Robotic Applications in Food Industry,” *International Journal of Robotics Applications and Technologies*, vol. 4, no. 1, pp. 43–62, Jan. 2016, doi: 10.4018/IJRAT.2016010103.
- [7] R. J. Moreno Masey, J. O. Gray, T. J. Dodd, and D. G. Caldwell, “Guidelines for the design of low-cost robots for the food industry,” *Industrial Robot: An International Journal*, vol. 37, no. 6, pp. 509–517, Oct. 2010, doi: 10.1108/01439911011081650.
- [8] SNS Insider, “Restaurant Delivery Robot Market to Surpass USD 63.82 billion by 2030 due to the growing demand for contactless dining and the need for efficient food delivery solutions.”

- [9] A. Tuomi, I. P. Tussyadiah, and J. Stienmetz, “Applications and Implications of Service Robots in Hospitality,” *Cornell Hospitality Quarterly*, vol. 62, no. 2, pp. 232–247, May 2021, doi: 10.1177/1938965520923961.
- [10] S. Ivanov and C. Webster, “Restaurants and robots: public preferences for robot food and beverage services,” *Journal of Tourism Futures*, vol. 9, no. 2, pp. 229–239, May 2023, doi: 10.1108/JTF-12-2021-0264.
- [11] F. Seyitoğlu, S. Ivanov, O. Atsız, and İ. Çifçi, “Robots as restaurant employees - A double-barrelled detective story,” *Technol Soc*, vol. 67, p. 101779, Nov. 2021, doi: 10.1016/j.techsoc.2021.101779.
- [12] A. Benabid Najjar, A. R. Al-Issa, and M. Hosny, “Dynamic indoor path planning for the visually impaired,” 2022, doi: 10.1016/j.jksuci.2022.03.004.
- [13] Stankovich Miriam and Gupta Ravi, “INDUSTRY 4.0 OPPORTUNITIES BEHIND THE CHALLENGE,” Vienna: UNIDO, Nov. 2017, p. 5.
- [14] M. A. K. Bahrin, M. F. Othman, N. H. N. Azli, and M. F. Talib, “Industry 4.0: A review on industrial automation and robotic,” *Jurnal Teknologi*, vol. 78, no. 6–13. Penerbit UTM Press, pp. 137–143, 2016. doi: 10.11113/jt.v78.9285.
- [15] Integral Innovation Experts, “UNA APROXIMACIÓN A LA INDUSTRIA 4.0 (INDUSTRY 4.0).”
- [16] D. Floreano, A. Martinoli, F. Mondada, and J. Nicoud, “DESIGN, CONTROL, AND APPLICATIONS OF AUTONOMOUS MOBILE ROBOTS,” 1999.
- [17] Sherpa Mobile Robotics, “¿Qué es un robot móvil autónomo?” Accessed: Nov. 12, 2023. [Online]. Available: <https://www.sherpa-mr.com/es/Un-robot-mobile-autonome-cest-quoi>
- [18] A. Loganathan and N. S. Ahmad, “A systematic review on recent advances in autonomous mobile robot navigation,” *Engineering Science and Technology, an International Journal*, vol. 40, p. 101343, Apr. 2023, doi: 10.1016/j.jestch.2023.101343.

- [19] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, "Introduction to autonomous mobile robots," 2011, Accessed: Oct. 18, 2023. [Online]. Available: <https://books.google.com/books?hl=es&lr=&id=4of6AQAAQBAJ&oi=fnd&pg=PP1&dq=roland+illah+r.+siegart+nourbakhsh+introduction+to+amr&ots=2x6d11pQG1&sig=mqz8pwJyCHmcScsiRyfs7sFE8PI>
- [20] Hokuyo, "A Guide to Selecting the Right LiDAR Sensors," *Robotics Tomorrow*, Dec. 2023.
- [21] "Leo Rover Blog - How is LiDAR used in Robotic Navigation? Pros and Cons." Accessed: Nov. 12, 2023. [Online]. Available: <https://www.leorover.tech/post/how-is-lidar-used-in-robotic-navigation-pros-and-cons>
- [22] C. Atwell, "ROS 2 Explained: Overview and Features," *Electronic Design*, Jan. 2022.
- [23] J. Kacprzyk, *Robot Operating System (ROS)*, vol. 895. Cham: Springer International Publishing, 2021. doi: 10.1007/978-3-030-45956-7.
- [24] N. Mishraa, D. Goyal, and A. D. Sharma, "Issues in Existing Robotic Service in Restaurants and Hotels," *SSRN Electronic Journal*, Apr. 2018, doi: 10.2139/SSRN.3166508.
- [25] "All-Robot Staff Serves, Cooks at China's Robot Restaurant - Eater." Accessed: Nov. 12, 2023. [Online]. Available: <https://www.eater.com/2012/6/28/6570185/all-robot-staff-serves-cooks-at-chinas-robot-restaurant>
- [26] Robotics 24/7, "Pudu Robotics Rolls Out PuduBot 2 Universal Delivery Robot - Robotics 24/7." Accessed: Jan. 24, 2024. [Online]. Available: [https://www.robotics247.com/article/pudu\\_robotics\\_rolls\\_out\\_pudubot\\_2\\_universal\\_delivery\\_robot](https://www.robotics247.com/article/pudu_robotics_rolls_out_pudubot_2_universal_delivery_robot)
- [27] M. Plank, C. Lemardelé, T. Assmann, and S. Zug, "Ready for robots? Assessment of autonomous delivery robot operative accessibility in German cities," *Journal of Urban Mobility*, vol. 2, p. 100036, 2022, doi: 10.1016/j.urbmob.2022.100036.

- [28] N. Sahal Alharbi, M. Yahia Qattan, and J. Haji Alhaji, "Towards Sustainable Food Services in Hospitals: Expanding the Concept of 'Plate Waste' to 'Tray Waste,'" *Sustainability*, vol. 12, no. 17, p. 6872, Aug. 2020, doi: 10.3390/su12176872.

## **Apéndices**

## Apéndice A

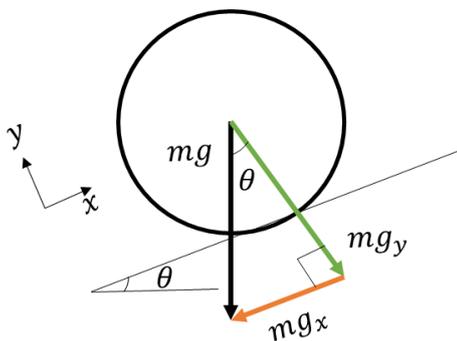
### Selección de motores

Para seleccionar adecuadamente el motor a utilizar, fue necesario realizar un análisis cinemático y dinámico para conocer el torque necesario del motor. El torque en un sistema de ruedas de un robot está directamente relacionado con la capacidad del motor para generar la fuerza necesaria para mover las ruedas y superar las resistencias al movimiento, como la fricción y la fuerza gravitacional. El peso del robot influye significativamente en el torque requerido. A medida que aumenta el peso del robot, también lo hace la fuerza gravitacional que actúa sobre él.

Para rodar en una superficie horizontal, el torque necesario para superar la fricción en las ruedas y cualquier pequeña irregularidad en la superficie es relativamente bajo. Sin embargo, cuando el robot se enfrenta a una pendiente, el peso del robot combinado con la gravedad crea una fuerza descendente que debe ser contrarrestada para mantener la velocidad constante.

**Figura A. 1**

*Diagrama de cuerpo libre 1*



$$mg_x = mg * \text{sen}(\theta) \quad (\text{A. 1})$$

$$mg_y = mg * \text{cos}(\theta) \quad (\text{A. 2})$$

El torque aplicado a las ruedas debe superar la resistencia de la fricción estática y dinámica entre las ruedas y la superficie sobre la que se mueven. Este torque se transmite desde el motor a través del sistema de transmisión (engranajes, ejes) hasta las ruedas. Para superar esta fricción y

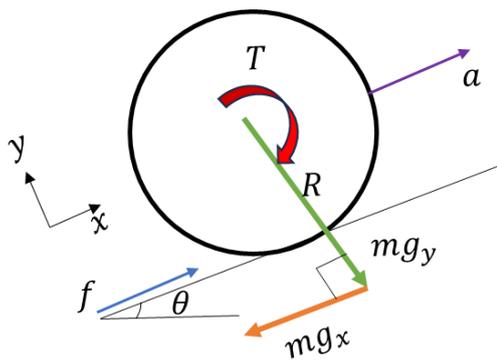
mover el robot, el torque debe ser suficiente para vencer la resistencia al giro de las ruedas. Dicho de otra forma, el torque requerido se expresa como:

$$T = f * R \quad (\text{A. 3})$$

Para el robot diseñado, se consideró el caso más crítico, en el cual el robot se desliza a través de una pendiente inclinada. Aplicando la segunda ley de Newton se tiene:

**Figura A. 2**

*Diagrama de cuerpo libre 2*



$$\sum F_x = M * a = f - mg_x \quad (\text{A. 4})$$

$$M * a = \frac{T}{R} - M * g * \text{sen}(\theta)$$

$$T = R * M * (a + g * \text{sen}(\theta))$$

Ahora, se divide el torque entre el número de ruedas motorizadas en el robot para poder distribuir la potencia generada por el motor de manera más equitativa entre las ruedas impulsadas y así mantener un equilibrio en la distribución de la fuerza generada por los motores. Para el caso del presente proyecto, se utilizaron 2 ruedas motorizadas.

$$T = \frac{R * M * (a + g * \text{sen}(\theta))}{N}$$

Finalmente, se considera la eficiencia del motor:

$$T = \left(\frac{100}{e}\right) * \frac{R * M * (a + g * \text{sen}(\theta))}{N}$$

Para el caso del robot se tienen los siguientes valores:

**Tabla A. 1 Valores iniciales para selección de motor**

Valores iniciales para selección de motor

<b>Masa (kg)</b>	<b>28</b>
<b>Número de ruedas motorizadas</b>	2
<b>Radio de las ruedas (m)</b>	0.07
<b>Inclinación máxima (grados)</b>	15
<b>Aceleración deseada (m/s<sup>2</sup>)</b>	0.2
<b>Eficiencia total (%)</b>	70

Para el valor de la masa, se tomó en cuenta el valor proporcionado por el software CAD Inventor una vez que se realizó el ensamble completo del robot y se especificó el material de cada una de las piezas. Este valor es de aproximadamente 18 kg. Adicionalmente, se suman los valores correspondientes a las cargas aplicadas a cada uno de los soportes de las bandejas, cada una con un valor máximo de 3 kg. Este valor se calculó de forma empírica en una balanza con una balanza y platos con comida en donde se tomó en cuenta un almuerzo típico, es decir, un plato de sopa, un plato de arroz, un vaso de jugo y un postre que puede ser una fruta. Así, el valor de la masa total del robot es de:

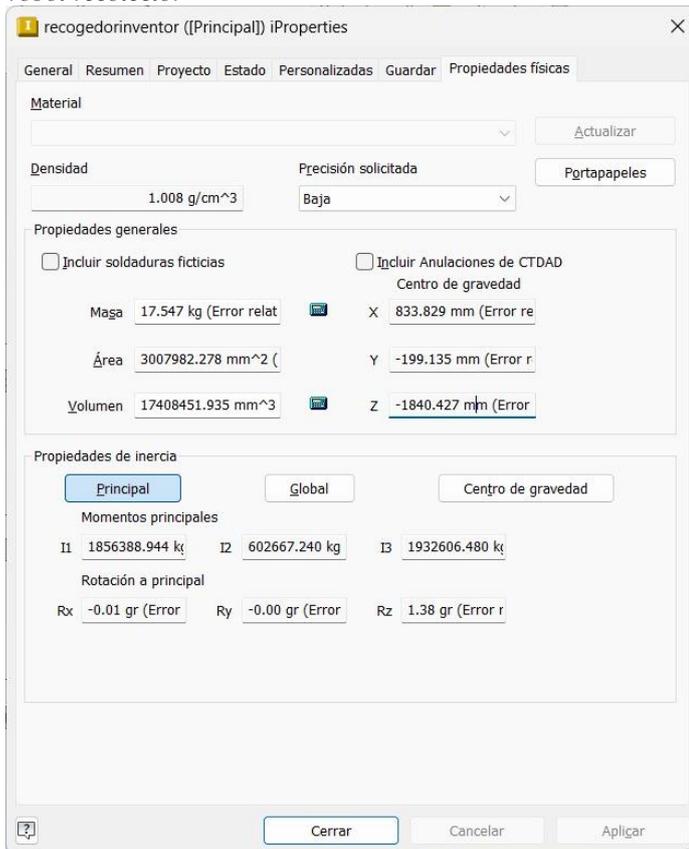
**Tabla A. 2**

Cálculo de masa aproximada del robot

Elemento	Masa
Masa de la estructura del robot (kg)	18
Masa de bandeja 1 (kg)	3
Masa de bandeja 2 (kg)	3
Masa de bandeja 3 (kg)	3
Otros elementos como: pantalla LCD, cables, tornillos, tuercas, etc (kg)	1
<b>Total (kg)</b>	<b>28</b>

**Figura A. 3**

*Masa de robot recolector*



Así, el torque aproximado que se calculó es de:

$$T = \left(\frac{100}{e}\right) * \frac{R * M * (a + g * \text{sen}(\theta))}{N}$$
$$T = \left(\frac{100}{70}\right) * \frac{0.07 * 28 * (0.2 + 9.81 * \text{sen}(15))}{2}$$

$$T = 3.83Nm$$

$$T = 39.1 \text{ kgf} - \text{cm}$$

Con ello, se seleccionó el motor que cumple con los valores más cercanos a los requerimientos mostrados. La elección de motores de corriente continua (DC) para un robot recolector de bandejas de comida rápida se basó en dos consideraciones clave:

1. **Control de velocidad y dirección:** Los motores DC ofrecen un control preciso sobre la velocidad y dirección del robot. Su capacidad para cambiar la polaridad de la corriente eléctrica permite el movimiento bidireccional y una amplia gama de velocidades, factores fundamentales para la navegación precisa en entornos de restaurantes o establecimientos de comida rápida.
2. **Torque y capacidad de arranque:** Los motores DC tienen una buena capacidad de torque, lo que resulta beneficioso al maniobrar bandejas o cargas. Esta característica es esencial para mover bandejas de comida sobre diferentes superficies.

En cuanto a motores con escobillas, su elección se fundamentó en la capacidad de proporcionar torque suficiente a bajas velocidades, lo cual se alinea con las necesidades de precisión y manipulación requeridas para interactuar con bandejas de comida. En este contexto, donde la prioridad recae en el control preciso y la capacidad de manipulación en lugar de la velocidad máxima, los motores con escobillas se ajustaron a una solución costo-efectiva más adecuada.

En contraste, los motores sin escobillas encuentran aplicación en robots que demandan alta velocidad y eficiencia energética, como en drones o vehículos aéreos no tripulados, debido a su diseño sin fricción y su capacidad para operar a altas RPM. Su uso está orientado a aplicaciones donde la velocidad extrema y la eficiencia son cruciales, aspectos que no son primordiales en el caso de un robot recolector de bandejas en un entorno de comida rápida, donde la precisión y el control son las principales prioridades.

**Tabla A. 3**

*Selección de motores Pololu*

Rated Voltage	Stall Current	No-Load Current	Gear Ratio	No-Load Speed (RPM)	Extrapolated Stall Torque		Max Power (W)	 Without Encoder	 With Encoder
					(kg · cm)	(oz · in)			
12 V	5.5 A	0.2 A	1:1 (no gearbox)	10,000	0.5	7	–	–	<a href="#">item #4750</a>
			6.3:1	1600	3.0	42	12	<a href="#">item #4747</a>	<a href="#">item #4757</a>
			10:1	1000	4.9	68	12	<a href="#">item #4748</a>	<a href="#">item #4758</a>
			19:1	530	8.5	120	12	<a href="#">item #4741</a>	<a href="#">item #4751</a>
			30:1	330	14	190	12	<a href="#">item #4742</a>	<a href="#">item #4752</a>
			50:1	200	21	290	10	<a href="#">item #4743</a>	<a href="#">item #4753</a>
			70:1	150	27	380	10*	<a href="#">item #4744</a>	<a href="#">item #4754</a>
			100:1	100	34	470	8*	<a href="#">item #4745</a>	<a href="#">item #4755</a>
			131:1	76	45	630	6*	<a href="#">item #4746</a>	<a href="#">item #4756</a>
150:1	67	49	680	6*	<a href="#">item #2829</a>	<a href="#">item #2828</a>			
24 V	3 A	0.1 A	1:1 (no gearbox)	10,000	0.55	8	–	–	<a href="#">item #4690</a>
			6.3:1	1600	3.5	49	14	<a href="#">item #4688</a>	<a href="#">item #4698</a>
			10:1	1000	5.5	76	14	<a href="#">item #4689</a>	<a href="#">item #4699</a>
			19:1	530	9.5	130	13	<a href="#">item #4681</a>	<a href="#">item #4691</a>
			30:1	330	15	210	13	<a href="#">item #4682</a>	<a href="#">item #4692</a>
			50:1	200	23	320	12	<a href="#">item #4683</a>	<a href="#">item #4693</a>
			70:1	140	31	430	10*	<a href="#">item #4684</a>	<a href="#">item #4694</a>
			100:1	100	39	540	8*	<a href="#">item #4685</a>	<a href="#">item #4695</a>
			131:1	79	47	650	6*	<a href="#">item #4686</a>	<a href="#">item #4696</a>
150:1	68	56	780	6*	<a href="#">item #4687</a>	<a href="#">item #4697</a>			

\* Output power for these units is constrained by gearbox load limits; spec provided is output power at max recommended load of 10 kg·cm.

## Potencia y corriente de motores

Es necesario conocer la potencia y la corriente que consumirán los motores de las ruedas. Para ello, se tiene que la potencia de un motor es:

$$P = T * \omega \quad (\text{A. 5})$$

Donde:

$$\omega = \frac{v}{r_{rueda}} \quad (\text{A. 6})$$

En la mayoría de las aplicaciones, estos robots tienen una velocidad lineal en un rango de 0.5 – 1.2 m/s, como es el caso del PuduBot. Por tanto, se tomará como referencia una velocidad de 0.5 m/s para el caso del robot recolector.

$$\omega = \frac{0.5}{0.07} \quad (\text{A. 7})$$

$$\omega = 7.14 \frac{rad}{s} * \frac{2\pi}{60}$$

$$\omega = 68.28 \text{ rpm}$$

Así,

$$P = \frac{T * \pi * n_{rpm}}{30} \quad (\text{A. 8})$$

$$P = \frac{3.83 * \pi * 68.28}{30}$$

$$P = 27.38 \text{ W}$$

$$P = 0.036 \text{ hp}$$

Por otro lado, según la ley de Ohm y potencia eléctrica, se tiene que:

$$P = I * V \quad (\text{A. 9})$$

$$I = \frac{P}{V}$$

$$I_{motor} = \frac{27.38}{12}$$

$$I_{motor} = 2.28 \text{ A}$$

## Apéndice B

### Selección de accesorios adicionales para el diseño mecánico

Figura B. 1

Par de soportes en L de aluminio estampado Pololu para motorreductores de metal de 37D mm

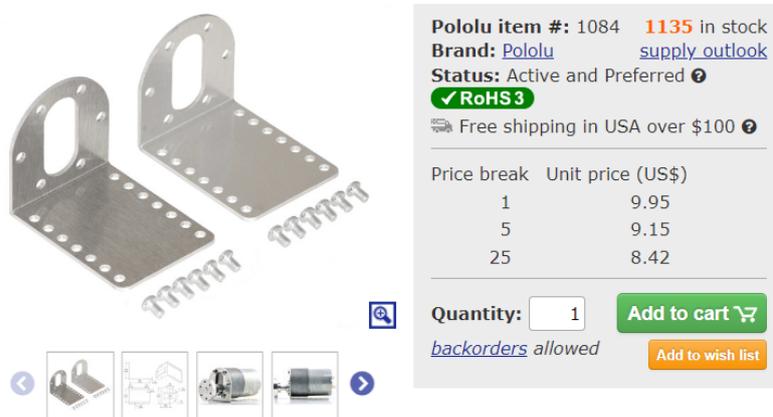


Figura B. 2

Adaptador de rueda de scooter de aluminio Pololu para eje de 6 mm



Este adaptador de aluminio ofrece una solución para adaptar las ruedas *scooter* seleccionadas con los motores cuyos ejes sean de 6 mm de diámetro.

Este accesorio cuenta con una pieza con orificios roscados y una arandela. La primera se fija al eje del motor mediante dos tornillos de fijación M4, garantizando su adaptabilidad a ruedas

con un espacio mínimo de 6 mm, donde generalmente se utilizan los rodamientos 608, mientras que la segunda se encarga de asegurar la rueda a la pieza mecanizada utilizando tres tornillos M3 de 14 mm (también incluidos en el paquete). Para mayor flexibilidad en la colocación de las ruedas en el eje, la arandela cuenta con un orificio central que opcionalmente permite el paso del eje de salida del motor.

## Apéndice C

### Selección de controlador principal

A continuación, se muestra una lista en forma de tabla comparativa de algunos de los controladores más utilizados en aplicaciones robóticas y sistemas embebidos, junto con varias especificaciones.

**Tabla C. 1**

*Tabla comparativa de especificaciones de controladores principales*

<b>Controlador / Especificación</b>	<b>Raspberry Pi 3A+</b>	<b>Raspberry Pi 3B+</b>	<b>Raspberry Pi 4B</b>	<b>Jetson nano dev</b>
<b>CPU</b>	1.4 GHz 64-bit Quad-core ARM Cortes-A53	1.4 GHz 64-bit Quad-core ARM Cortes-A53	1.5 GHz 64-bit Quad-core ARM72	1.4 GHz 64- bit Quad-Core ARM Cortex- A57 MPCore
<b>RAM</b>	512 MB LPDDR2 SDRAM	1GB LPDDR2 SDRAM	1GB, 2GB,4GB u 8GB	4GB LPDDR4
<b>GPIO</b>	40-pin	40-pin	40-pin	40-pin
<b>Dimensiones</b>	65 x 56 mm	85 x 56 mm	85 x 53 mm	100 x 79 mm
<b>Puertos</b>	1x USB 2.0	4x USB 2.0	2x USB 3.0 2x USB 2.0	4x USB 3.0
<b>Alimentación</b>	5 V	5 V	5 V	5V
<b>Precio</b>	\$25	\$35	\$85	\$99



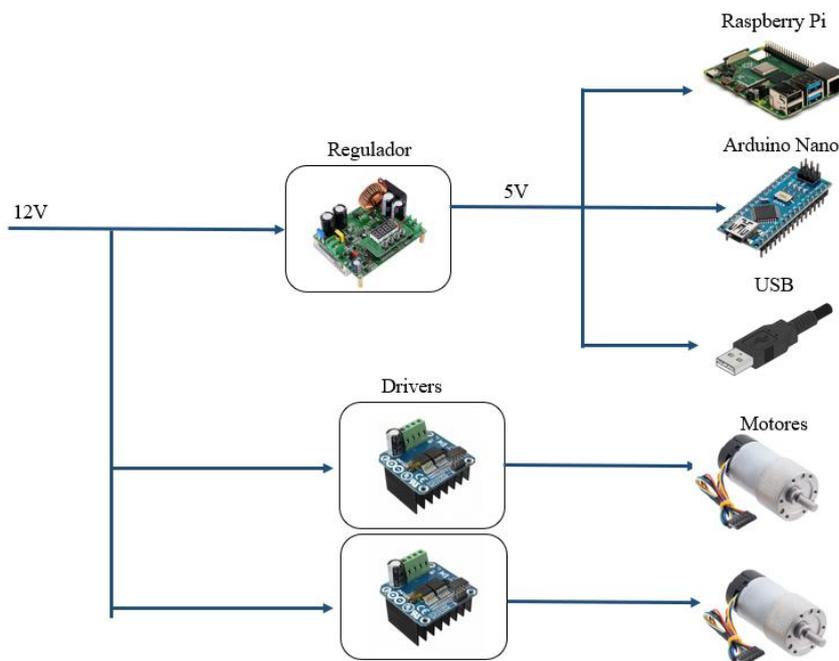
## Apéndice D

### Selección de la batería

Luego de seleccionar los componentes electrónicos para el robot, se realizó el cálculo de la corriente estimada para los componentes de 5V. Asimismo, se seleccionó un regulador de voltaje como suministro para dichos equipos. Sin embargo, también se requiere alimentar a los motores DC del robot, para lo cual, es necesario una batería. La selección del voltaje de la batería se realizó considerando las demandas específicas de alimentación de los componentes del robot, en particular, los motores de DC que requieren una fuente de 12 V para su funcionamiento óptimo. Para resumir los voltajes utilizados en el robot, se presenta la Figura E. 1 a continuación:

**Figura D. 1**

Resumen de alimentación de componentes a 12 V y 5 V



En la Tabla 2.12 se determinó que el regulador consumirá un total de 25 W de potencia, esto se respalda mediante la ecuación de potencia:

$$P = V * I \quad (\text{D. 1})$$

$$P = 5 * 5$$

$$P = 25 \text{ W}$$

Si se considera toda la parte de 5 V como un único componente de 25 W, y se intenta obtener exactamente 5 A a 5 V, siendo el regular 100 % eficiente con una entrada de 12 V, se usaría alrededor de 2.5 A de esa fuente. Las pérdidas de eficiencia y fluctuaciones en la batería llevan a redondear este número.

$$I = \frac{P}{V} \quad (\text{D. 2})$$

$$I = \frac{25}{12}$$

$$I = 2.5 \text{ A}$$

Dicha corriente corresponde a los componentes que trabajan a 5 V. En cuanto a los motores, el consumo de corriente depende de su tipo y carga. En la Tabla A. 3 se muestran las especificaciones del motor seleccionado, en el cual se indica dicha información. Por ejemplo, la hoja de datos muestra una corriente máxima de 5.5 A y una corriente nominal de 0.2 A. Como se tiene 2 motores:

$$I_{\text{max\_motores}} = 5.5 * 2$$

$$I_{\text{max\_motores}} = 11 \text{ A}$$

De esta forma, se estableció la corriente total requerida por la batería de 12 V mediante la suma de la corriente consumida por los componentes de 5 V y los motores de 12 V:

$$I_{\text{max\_robot}} = 2.5 + 11$$

$$I_{\text{max\_robot}} = 13.5 \text{ A}$$

Con este valor, se procedió a seleccionar una batería que cumpla básicamente 2 criterios:

- 12 V aproximadamente en su salida
- Capaz de manejar al menos 13.5 A

El candidato ideal fue una batería *LiPO* de 3 celdas.

Las baterías *LiPo* son baterías de polímero de litio, conocidas por su alta densidad de energía y capacidad de proporcionar un suministro de energía constante para dispositivos como robots, drones y otros equipos electrónicos.

En cuanto a las principales especificaciones que se encuentran en las baterías *LiPo*, suelen indicar:

- **Número de celdas:** Las baterías *LiPo* suelen tener un número seguido de "S", que indica el número de celdas conectadas en serie. La batería seleccionada tiene una configuración 3S, es decir, tres celdas *LiPo* conectadas en serie. Cada celda tiene un voltaje nominal de alrededor de 3.7 voltios, por lo que tres celdas en serie producen un voltaje total de:

$$V_{total} = 3 \text{ celdas} * 3.7 \text{ voltios por celda} \quad (\text{D. 3})$$

$$V_{total} = 11.1 \text{ V}$$

- **Capacidad:** La capacidad de la batería se expresa en miliamperios-hora (mAh) o amperios-hora (Ah). Esto indica cuánta energía puede almacenar la batería. Para este proyecto, la capacidad se puede calcular de la siguiente forma:

$$C_{batería} = \frac{7000mAh}{1000} = 7Ah \quad (\text{D. 4})$$

$$T_{duración} = \frac{C_{batería}}{I_{max\_robot}} = \frac{7Ah}{13.5A} = 0.518 \text{ h}$$

$$T_{duración} = 0.518 \text{ h} * 60$$

$$T_{duración} = 31.11 \text{ min}$$

Por lo tanto, con la batería seleccionada de 7000mAh y el consumo del robot, de 13.5A, la batería duraría aproximadamente 31.11 minutos.

- **Tasa de descarga:** A menudo, consiste en un número seguido de una letra "C" (por ejemplo, 25C, 50C). Esto indica la capacidad de descarga máxima continua que la batería puede proporcionar de manera segura. Por ejemplo, en este caso la batería seleccionada es

de 7000 mAh con una tasa de descarga de 100C y puede suministrar una corriente máxima continua de 700 amperios de forma segura ( $7000\text{mAh} * 100\text{C} / 1000$ ).

Ahora, cabe recalcar que dicha capacidad de batería se calculó utilizando la corriente máxima de los motores, es decir, la corriente de paro, cuya especificación técnica es de 5.5 A, sin embargo, en el Apéndice A, se calculó la corriente de los motores en sus condiciones normales de operación. Dicho valor obtenido fue de 2.28 A por motor, es decir 4.56 A por los 2 motores. Por tanto, se tiene que en condiciones normales:

$$C_{batería} = \frac{7000\text{mAh}}{1000} = 7\text{Ah}$$

$$T_{duración} = \frac{C_{batería}}{I_{robot}} = \frac{7\text{Ah}}{4.56\text{A}} = 1.535\text{ h}$$

$$T_{duración} = 1.535\text{ h} * 60$$

$$T_{duración} = 92.1\text{ min}$$

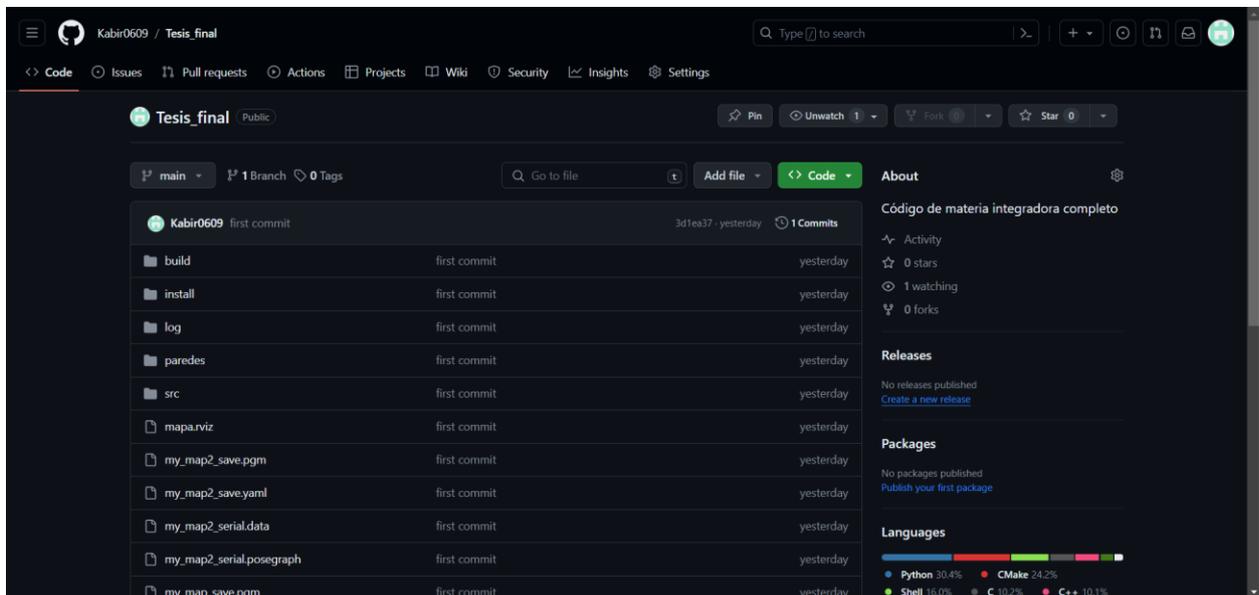
$$T_{duración} = 1.53\text{ h}$$

Es decir que, en condiciones normales, el robot podrá operar durante aproximadamente 1.53 h.

## Apéndice E

### Repositorio GitHub

[https://github.com/Kabir0609/Tesis\\_final.git](https://github.com/Kabir0609/Tesis_final.git)



## **Apéndice F**

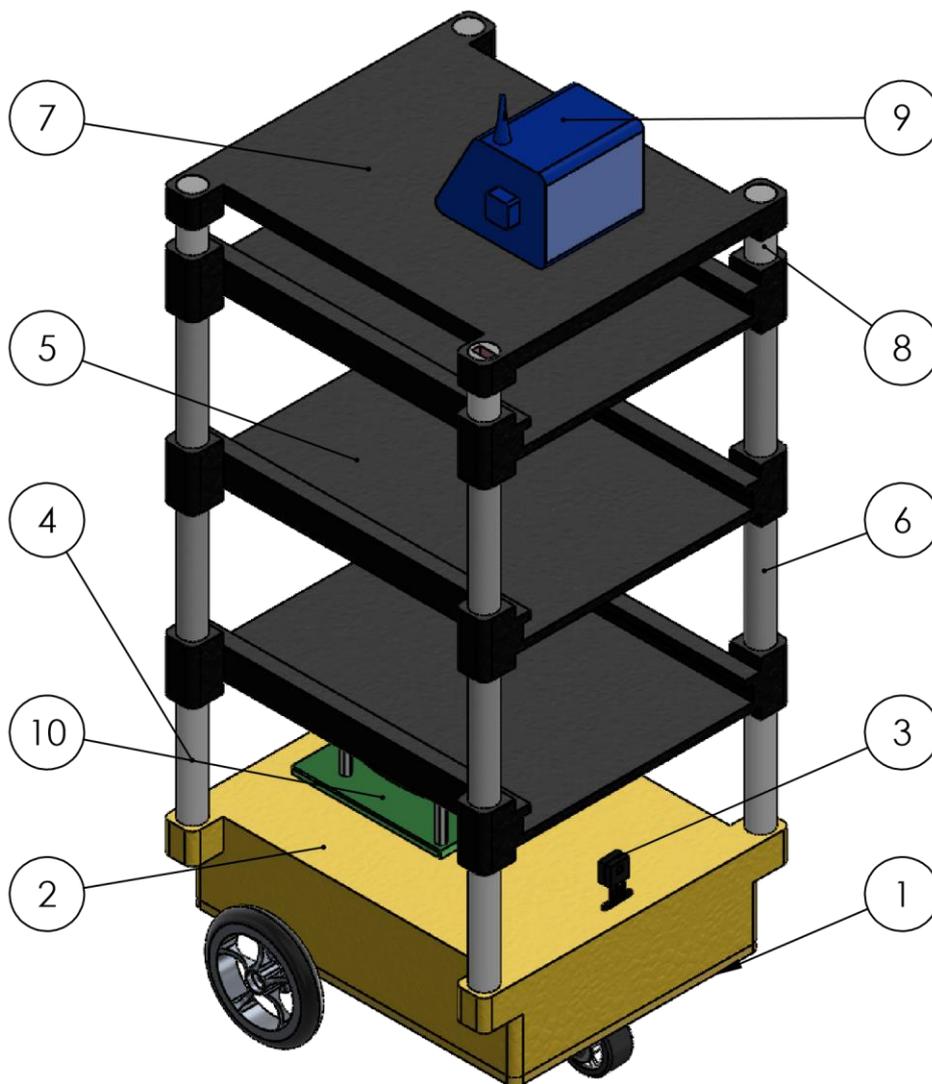
### **Planos de piezas mecánicas**

1

2

3

4



ITEM No	DESCRIPCIÓN	CANTIDAD
1	Base chasis	1
2	Parte superior del chasis	1
3	Soporte cámara Pi	1
4	Unión chasis - bandeja	4
5	Soporte de Bandeja	3
6	Unión bandeja - bandeja	8
7	Soporte cabeza	1
8	Unión bandeja - cabeza	4
9	Soporte de pantalla	1
10	Soporte sensor Lidar	1

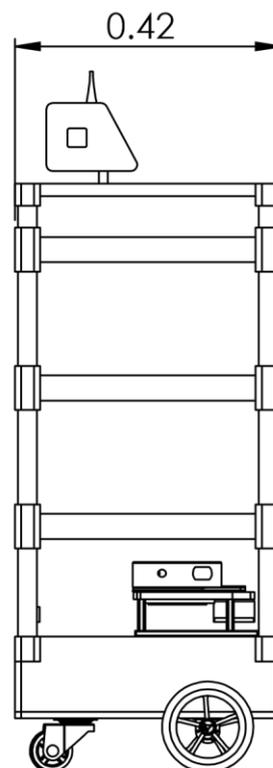
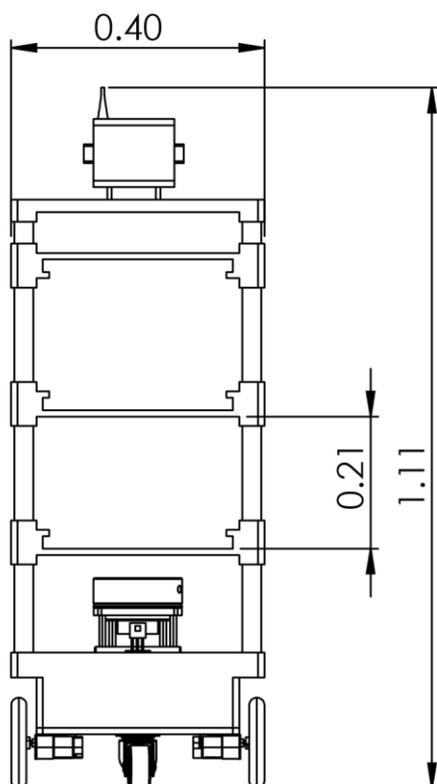
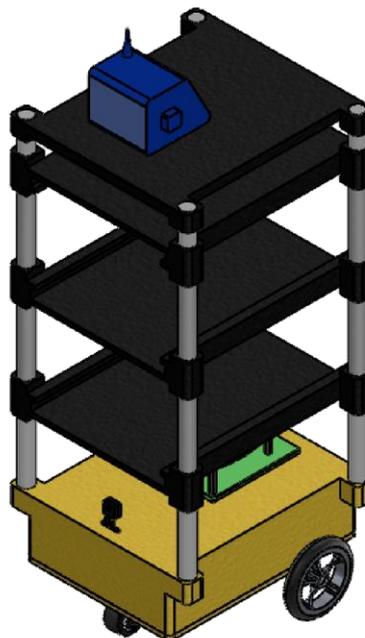
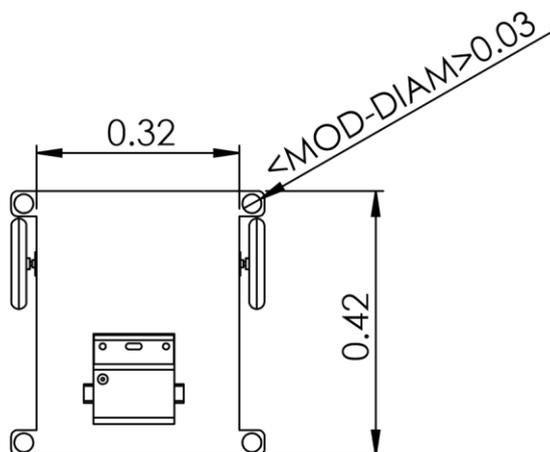
	NOMBRE	DIBUJO	FECHA
DIBUJADO	Juan Carlos Yépez	<b>ESTRUCTURA MECÁNICA (PARTES)</b>	10/1/2024
DIBUJADO	Kabir Cañarte		REVISIÓN:
TUTOR	Carlos Saldarriaga	FACULTAD	FIMCP
HOJA	<b>A4</b>	escuela superior Politécnica del Litoral	ESCALA <b>1:8</b>
			Nº: <b>1</b>
			UNIDAD <b>mm</b>

1

2

3

4



A

B

C

D

E

F

NOMBRE

DIBUJO

FECHA

DIBUJADO

Juan Carlos Yépez

## MEDIDAS GENERALES DEL ROBOT

10/1/2024

DIBUJADO

Kabir Cañarte

REVISIÓN:

TUTOR

Carlos Saldarriaga

FACULTAD

FIMCP



HOJA

A4

**espol** Escuela Superior  
Politécnica del Litoral

ESCALA

1:12

N°:

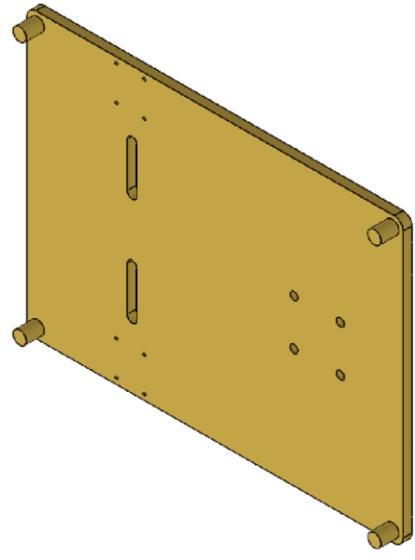
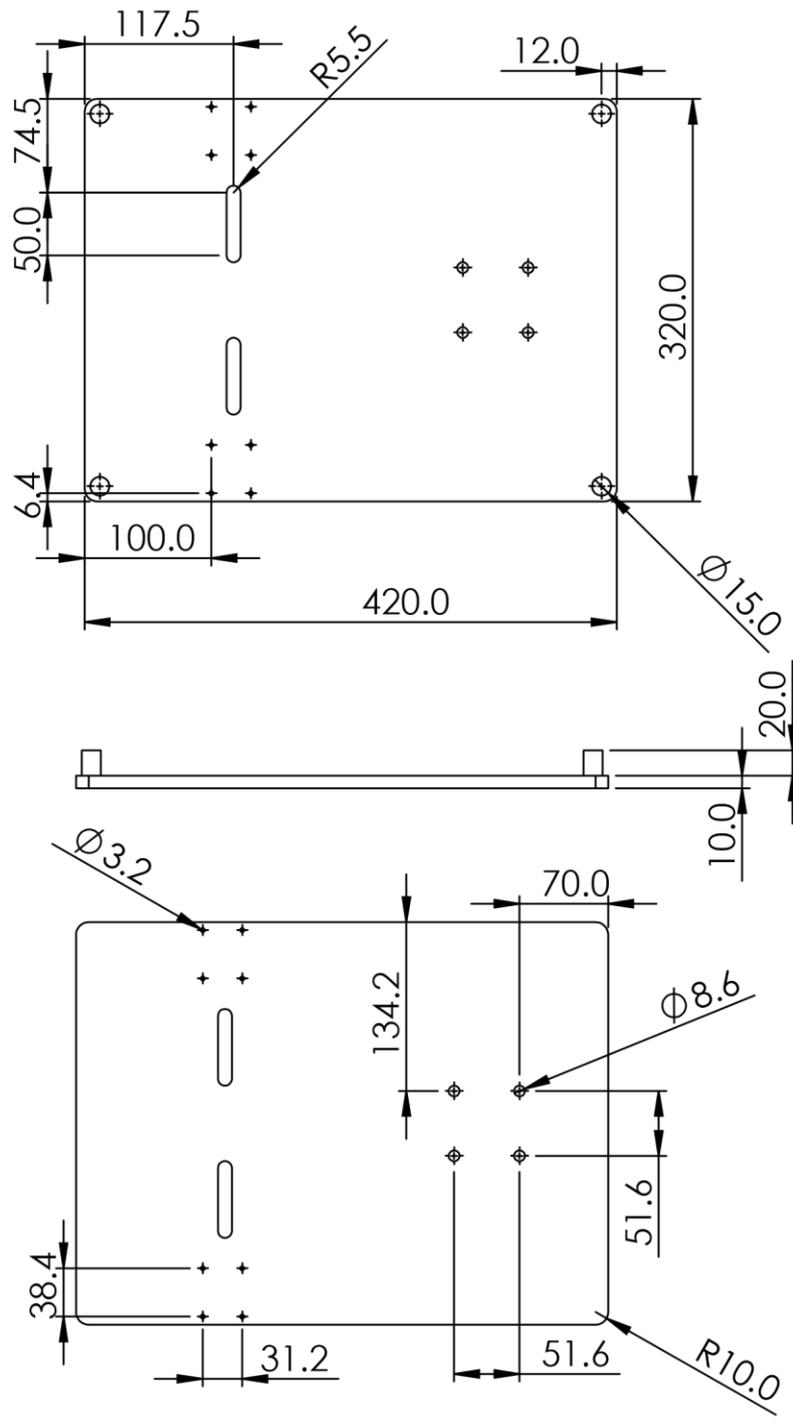
2

UNIDAD

mm

1 2 3 4

A  
B  
C  
D  
E  
F



	NOMBRE	DIBUJO	FECHA
DIBUJADO	Juan Carlos Yépez	<b>BASE DEL CHASIS</b>	10/1/2024
DIBUJADO	Kabir Cañarte		REVISIÓN:
TUTOR	Carlos Saldarriaga	FACULTAD	<b>FIMCP</b>
	HOJA	ESCALA	Nº: UNIDAD
	<b>A4</b>		<b>1:6</b> <b>2</b> <b>mm</b>

1

2

3

4

A

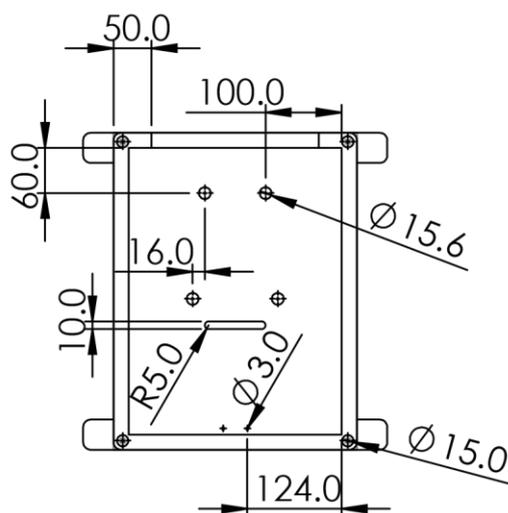
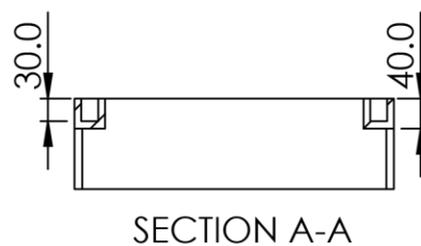
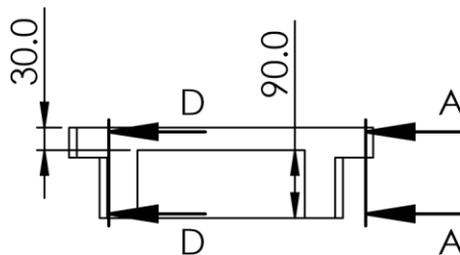
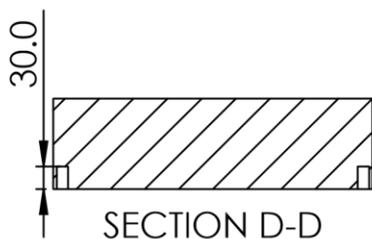
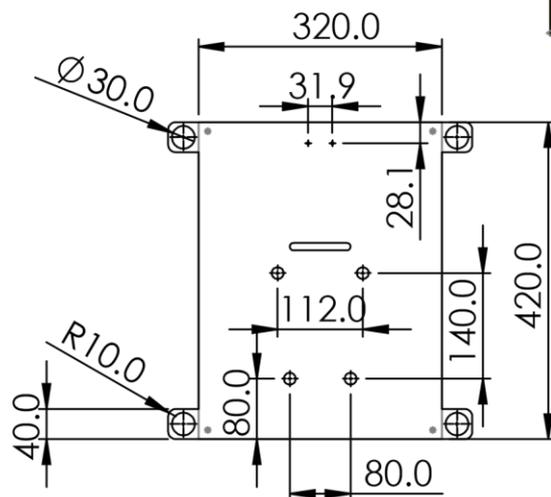
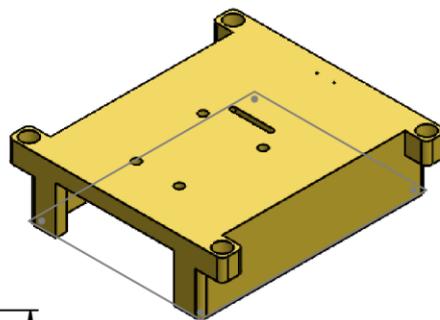
B

C

D

E

F



NOMBRE

DIBUJO

FECHA

DIBUJADO

Juan Carlos Yépez

**PARTE SUPERIOR DE CHASIS**

10/1/2024

DIBUJADO

Kabir Cañarte

REVISIÓN:

TUTOR

Carlos Saldarriaga

FACULTAD

**FIMCP**

HOJA

**A4**

ESCALA

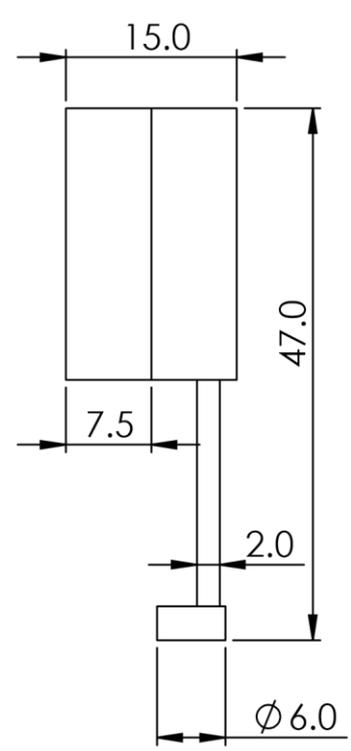
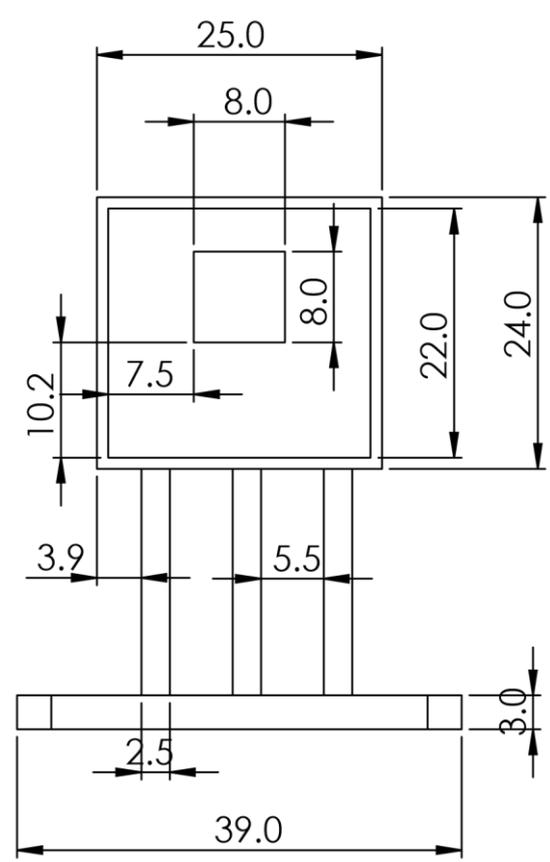
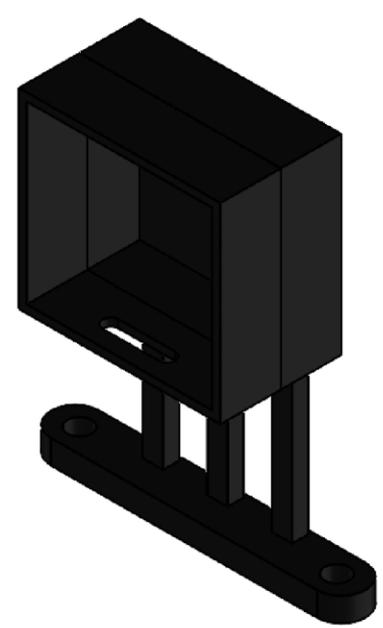
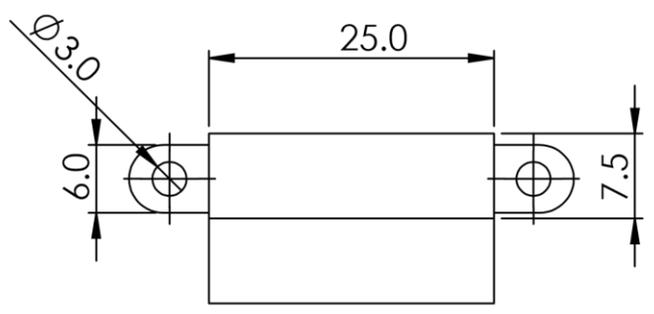
**1:10**

N°:

**3**

UNIDAD

**mm**



	NOMBRE	DIBUJO	FECHA		
DIBUJADO	Juan Carlos Yépez	<b>SOPORTE PARA CÁMARA PI</b>	10/1/2024		
DIBUJADO	Kabir Cañarte		REVISIÓN:		
TUTOR	Carlos Saldarriaga	FACULTAD		FIMCP	
	HOJA		ESCALA	N°:	UNIDAD
	<b>A4</b>		<b>3:2</b>	<b>4</b>	<b>mm</b>

1

2

3

4

A

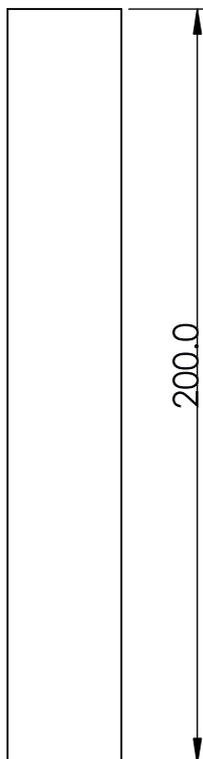
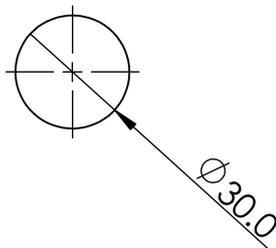
B

C

D

E

F



	NOMBRE	DIBUJO		FECHA	
DIBUJADO	Juan Carlos Yépez	<b>UNIÓN CHASIS - BANDEJA</b>		10/1/2024	
DIBUJADO	Kabir Cañarte			REVISIÓN:	
TUTOR	Carlos Saldarriaga			FACULTAD <b>FIMCP</b>	
	HOJA <b>A4</b>			ESCALA <b>1:2</b>	Nº: <b>5</b>

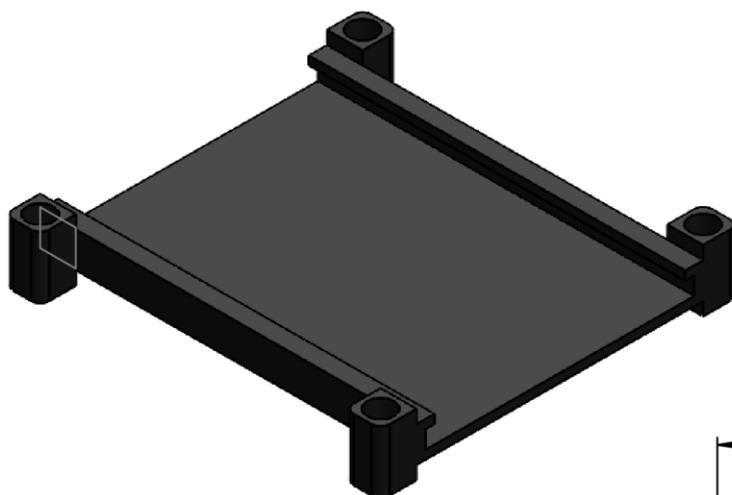
1

2

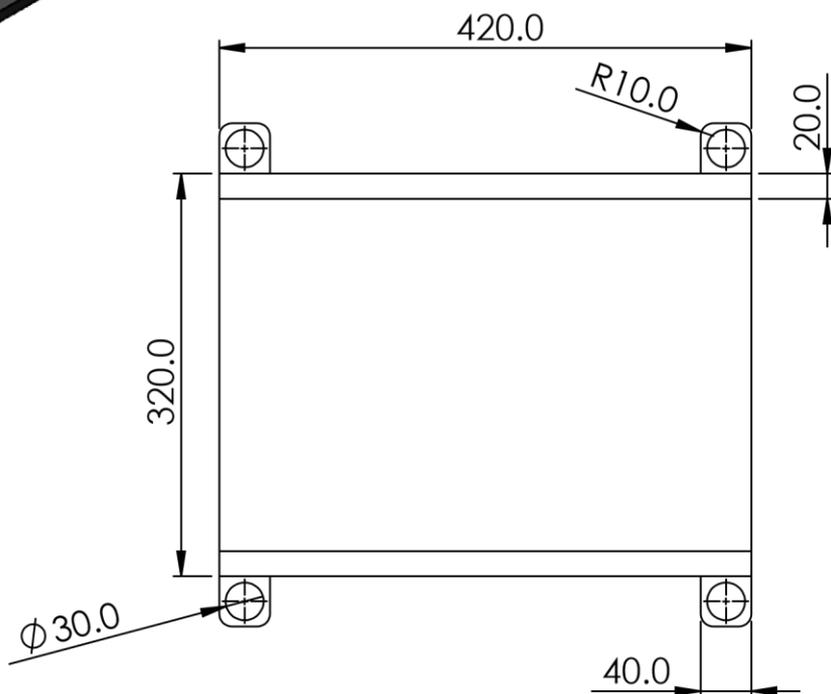
3

4

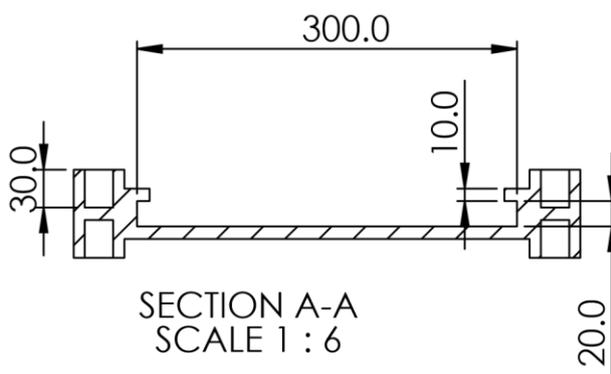
A



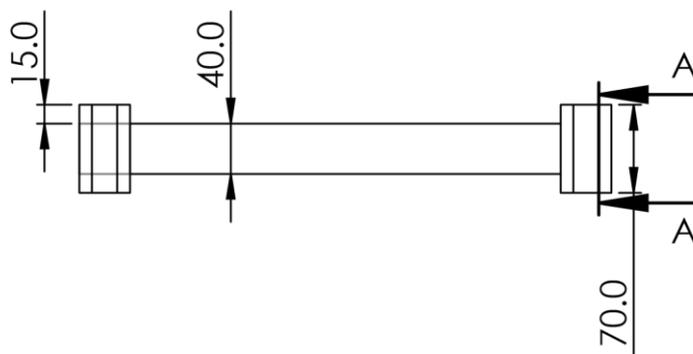
B



C



D



E

F

	NOMBRE	DIBUJO	FECHA
DIBUJADO	Juan Carlos Yépez	<b>SOPORTE DE BANDEJA</b>	10/1/2024
DIBUJADO	Kabir Cañarte		REVISIÓN:
TUTOR	Carlos Saldarriaga	FACULTAD	<b>FIMCP</b>
	HOJA	ESCALA	Nº: UNIDAD
	<b>A4</b>		<b>1:6</b> <b>6</b> <b>mm</b>

1

2

3

4

A

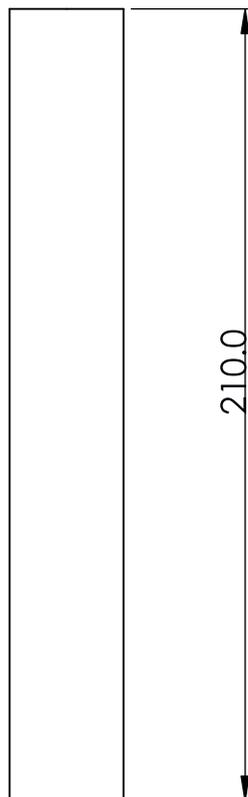
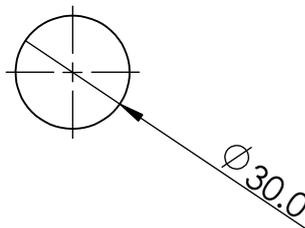
B

C

D

E

F



	NOMBRE	DIBUJO	FECHA		
DIBUJADO	Juan Carlos Yépez	<b>UNIÓN BANDEJA - BANDEJA</b>	10/1/2024		
DIBUJADO	Kabir Cañarte		REVISIÓN:		
TUTOR	Carlos Saldarriaga	FACULTAD		<b>FIMCP</b>	
	HOJA	<b>espol</b> Escuela Superior Politécnica del Litoral	ESCALA	Nº:	UNIDAD
	<b>A4</b>		<b>1:2</b>	<b>7</b>	<b>mm</b>

1

2

3

4

A

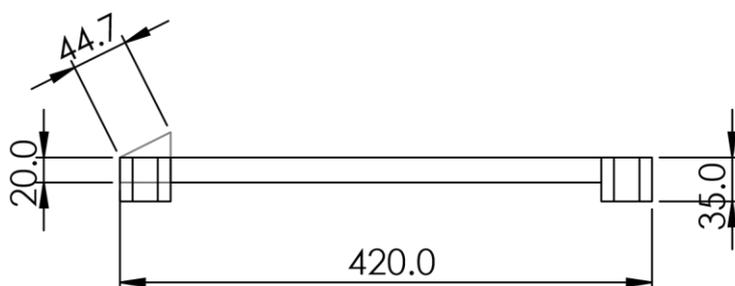
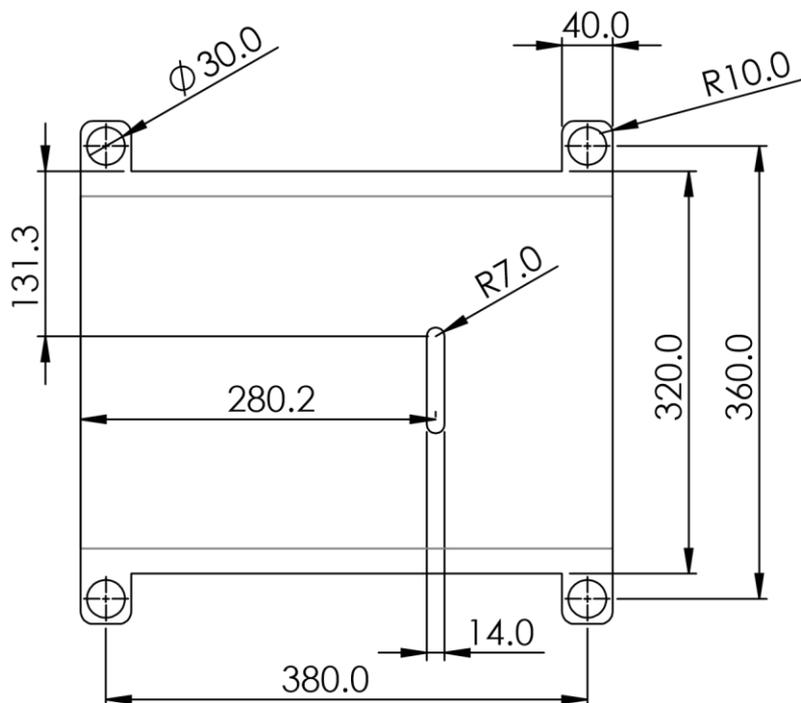
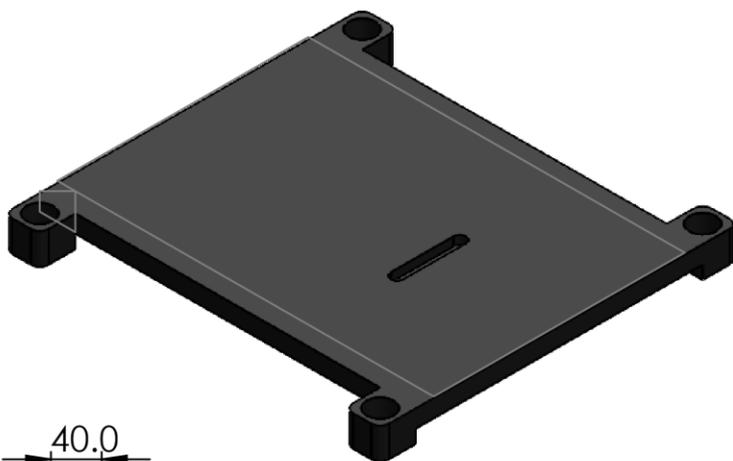
B

C

D

E

F



	NOMBRE	DIBUJO	FECHA
DIBUJADO	Juan Carlos Yépez	<b>SOPORTE BANDEJA - CABEZA</b>	10/1/2024
DIBUJADO	Kabir Cañarte		REVISIÓN:
TUTOR	Carlos Saldarriaga	FACULTAD	<b>FIMCP</b>
	HOJA		ESCALA
	<b>A4</b>		<b>1:6</b>
			UNIDAD
			<b>mm</b>

1

2

3

4

A

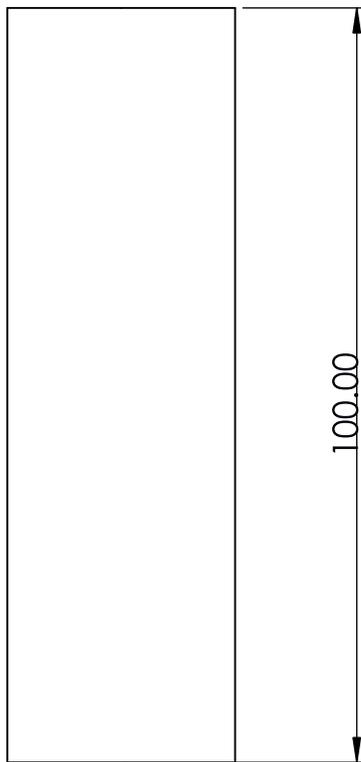
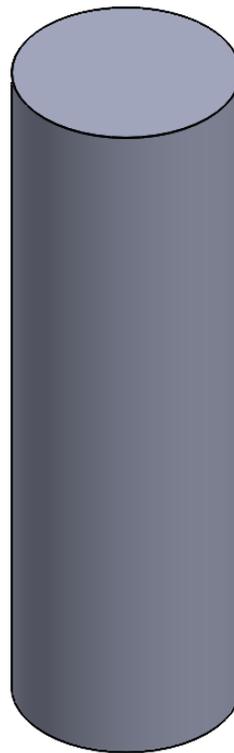
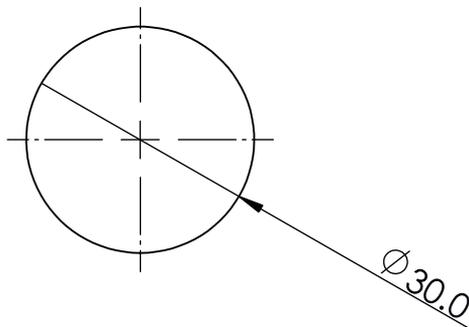
B

C

D

E

F



	NOMBRE	DIBUJO	FECHA
DIBUJADO	Juan Carlos Yépez	<b>UNIÓN BANDEJA-CABEZA</b>	10/1/2024
DIBUJADO	Kabir Cañarte		REVISIÓN:
TUTOR	Carlos Saldarriaga	FACULTAD	<b>FIMCP</b>
	HOJA	Escuela Superior Politécnica del Litoral	ESCALA
	<b>A4</b>		<b>1:1</b>
			UNIDAD
			<b>mm</b>

1

2

3

4

A

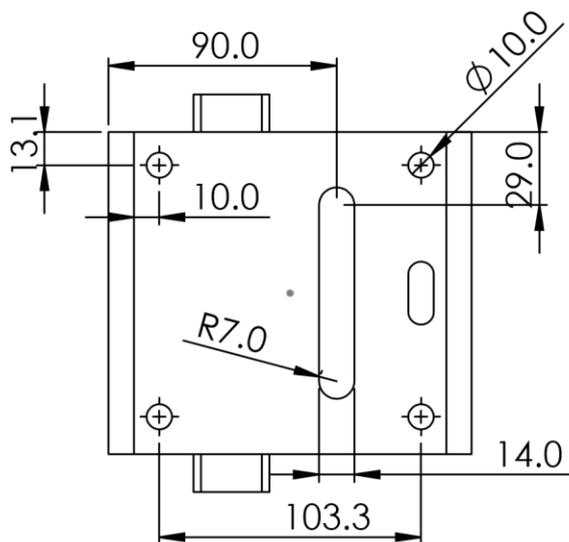
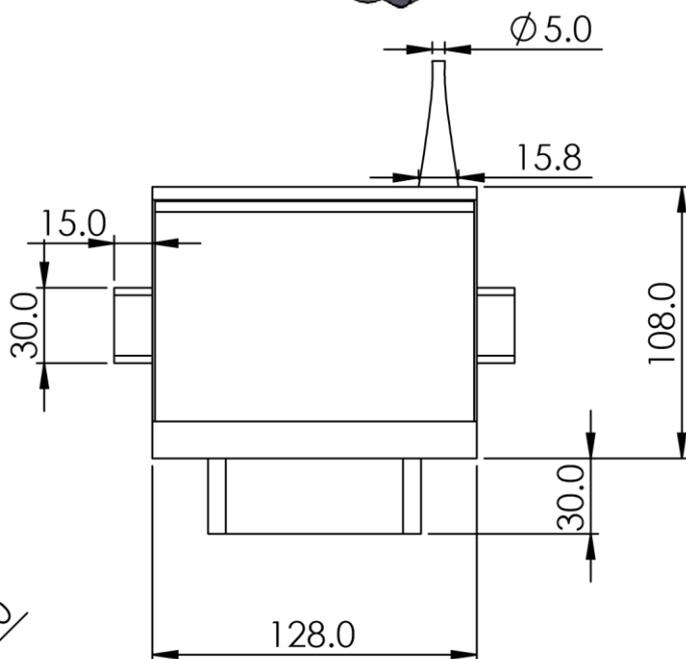
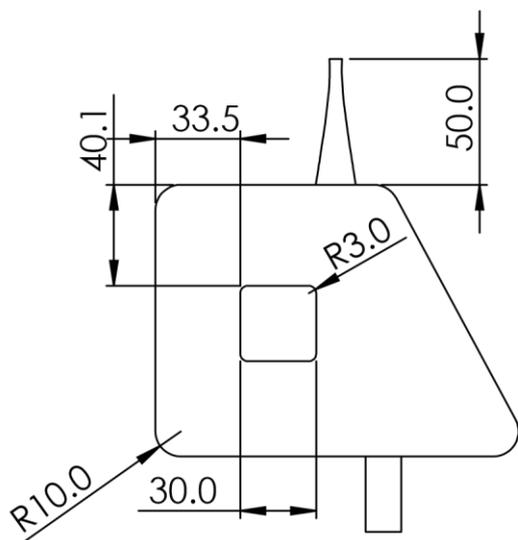
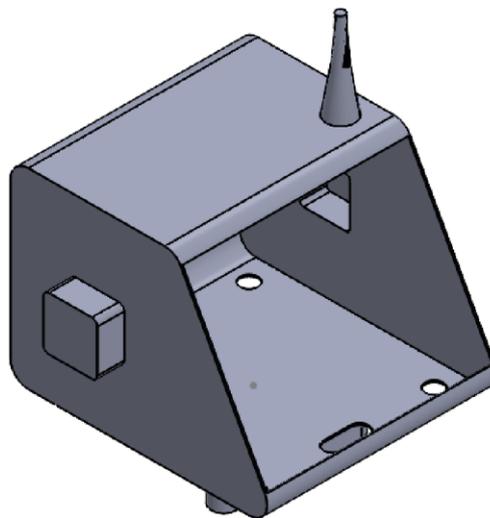
B

C

D

E

F



NOMBRE

DIBUJO

FECHA

DIBUJADO

Juan Carlos Yépez

**SOPORTE DE PANTALLA**

10/1/2024

DIBUJADO

Kabir Cañarte

REVISIÓN:

TUTOR

Carlos Saldarriaga

FACULTAD

**FIMCP**

HOJA

**A4**

ESCALA

**1:3**

N°:

**10**

UNIDAD

**mm**

1

2

3

4

A

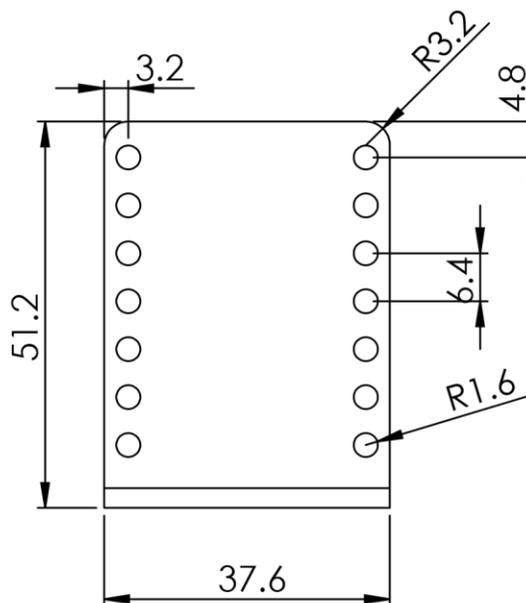
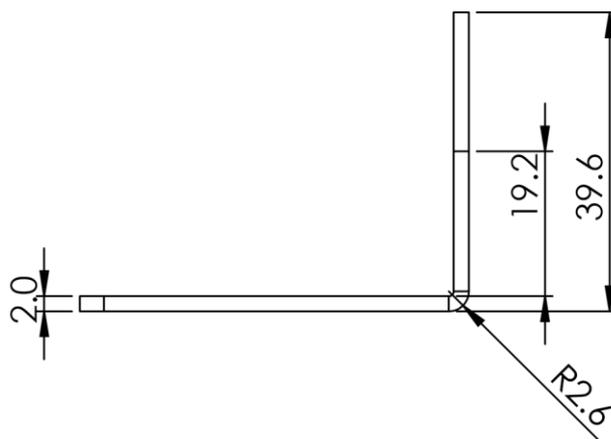
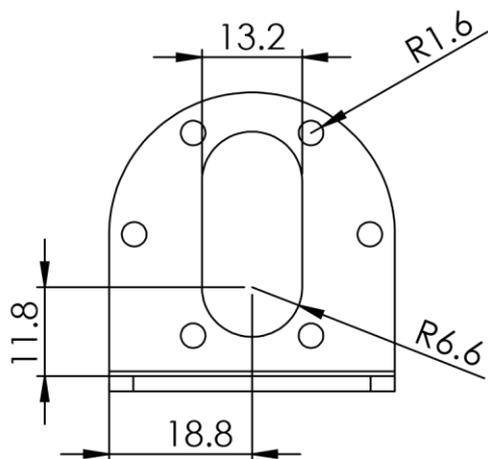
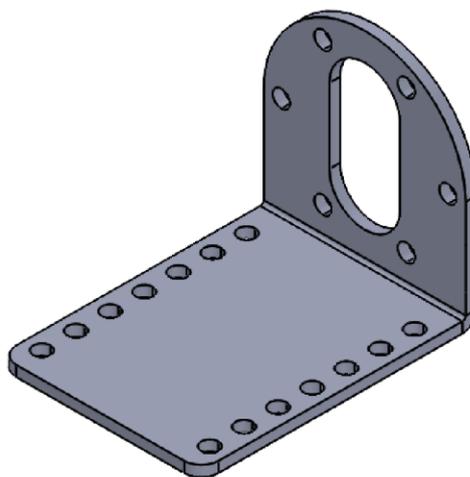
B

C

D

E

F



	NOMBRE	DIBUJO	FECHA
DIBUJADO	Juan Carlos Yépez	<b>SOPORTE MOTOR-CHASIS</b>	10/1/2024
DIBUJADO	Kabir Cañarte		REVISIÓN:
TUTOR	Carlos Saldarriaga	FACULTAD	<b>FIMCP</b>
	HOJA		ESCALA
	<b>A4</b>		<b>1:1</b>