

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

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

Implementación de un software de diseño mecatrónico para el
dimensionamiento y modelado 3D de drones UAV multipropósito

PROYECTO INTEGRADOR

Previo la obtención del Título de:

Ingenieros en Mecatrónica

Presentado por:

Hansell Dimaneth Jaramillo Granda

Arom Moises Moreno Ortiz

GUAYAQUIL - ECUADOR

Año: 2022

DEDICATORIA

Para aquellos que han sido parte de mi viaje y me han dado la motivación y fuerza para seguir adelante, esta tesis es un homenaje a su constante apoyo y dedicación. Agradezco de todo corazón su presencia en mi vida.

Arom Moises Moreno Ortiz

A mi padre y madre, quienes me han sabido guiarme sabiamente en el camino de la vida y han sido esenciales en mi formación académica, a mis hermanos y enamorada, que siempre han estado apoyándome en todo momento y nunca han dejado de creer en mí.

Hansell Dimaneth Jaramillo Granda

AGRADECIMIENTOS

Agradezco de corazón a aquellos que han hecho posible la realización de esta tesis. En especial a mis profesores y tutores, quienes, con su experiencia, sabiduría y guía incansable, me han brindado las herramientas necesarias para alcanzar mis metas. También quiero agradecer a mis colegas y amigos, quienes han sido una fuente constante de motivación y apoyo emocional. Gracias por formar parte de mi camino académico y profesional.

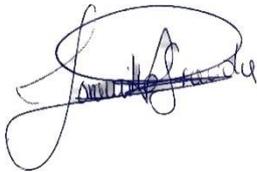
Arom Moises Moreno Ortiz

Agradezco a Dios por todas las formas en que me ha bendecido para poder llegar hasta aquí, a mi familia por siempre estar presente, al Ing. Christian Tutivén por ser nuestro tutor, guía y apoyo en el presente trabajo. A mis compañeros y amigos que he hecho a lo largo de mi vida académica, de los cuales he aprendido y por los que estaré eternamente agradecido.

Hansell Dimaneth Jaramillo Granda

DECLARACIÓN EXPRESA

"Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; Hansell Jaramillo Granda y Arom Moreno Ortiz, damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"



Hansell Dimaneth
Jaramillo Granda



Arom Moises Moreno
Ortiz

EVALUADORES

.....
Carlos Saldarriaga Ph.D.

PROFESOR DE LA MATERIA

.....
Christian Tutivén Ph.D.

PROFESOR TUTOR

RESUMEN

Los drones o VANT (Vehículos aéreo son equipos tecnológicos innovadores y relativamente nuevos, surgieron de la necesidad de mejorar la eficiencia y efectividad en diferentes ámbitos, como la exploración, la seguridad, la agricultura, la publicidad, entre otros. Aunque el surgimiento de los drones se puede considerar un producto de la cuarta revolución industrial, los primeros prototipos y conceptos de estos vehículos aéreos no tripulados existieron desde la década de 1960 [1]. Los avances tecnológicos en el campo de los drones han permitido la creación de diferentes tipos de estos dispositivos, como los drones de tipo cópter y los de tipo avión, cada uno con sus propias ventajas y desventajas dependiendo la aplicación a desarrollar entre otros factores [2].

El presente proyecto tiene como objetivo desarrollar un software para el dimensionamiento, optimización y diseño de un VANT de tipo cuadricóptero, utilizando los conocimientos en programación y basándose en el software de código abierto OpenScad. Se construyó un algoritmo para el dimensionamiento del VANT mediante el uso de ecuaciones basadas en la teoría de construcción de drones y se programaron funciones de generación automática para el modelado 3D de las partes del fuselaje del dron. Durante el desarrollo, se utilizaron técnicas de programación funcional para crear un prototipo virtual, el cual fue validado mediante el modelado 3D y el ensamblaje correcto de las piezas, y se visualizó el producto en otros softwares de diseño 3D.

En conclusión, el proyecto logró desarrollar un software que permite el dimensionamiento, optimización y diseño de un VANT de tipo cuadricóptero de manera eficiente y basada en conocimientos en programación.

Palabras Clave: Python, OpenScad, software, fuselaje, VANT

ABSTRACT

Drones or UAVs (Unmanned Aerial Vehicles) are innovative and relatively new technology that emerged as a result of the fourth industrial revolution. Currently, they are used in different professional activities, such as aerial photogrammetry, advertising purposes and agrochemical tasks. Technological advancements in the field of drones have allowed for the creation of different types of these devices, such as copter and plane types, each with their own advantages and disadvantages.

The objective of this project is to develop software for sizing, optimization, and design of a quadcopter type UAV, utilizing programming knowledge and based on the open-source software OpenScad. An algorithm was built for the sizing of the UAV using equations based on drone construction theory and automatic generation functions were programmed for 3D modeling of the drone fuselage parts. During development, functional programming techniques were used to create a virtual prototype, which was validated through 3D modeling and correct assembly of the parts, and the product was visualized in other 3D design software.

ÍNDICE GENERAL

RESUMEN.....	I
ABSTRACT	II
ÍNDICE GENERAL	III
ABREVIATURAS.....	V
SIMBOLOGÍA	VI
ÍNDICE DE TABLAS.....	VII
ÍNDICE DE FIGURAS	VIII
ÍNDICE DE PLANOS.....	X
CAPÍTULO 1	
1. INTRODUCCIÓN	11
1.1 DESCRIPCIÓN DEL PROBLEMA.....	12
1.2 JUSTIFICACIÓN DEL PROBLEMA	13
1.3 OBJETIVOS	14
1.3.1 Objetivo General.....	14
1.3.2 Objetivos Específicos	14
1.4 MARCO TEÓRICO	14
1.4.1 Componentes de un dron	14
1.4.2 Programación OpenScad	17
1.4.3 Lenguaje de programación	19
1.4.4 Estado del arte.....	20
CAPÍTULO 2	
2. METODOLOGÍA.....	23
2.1 SELECCIÓN DE LA ALTERNATIVA DE SOLUCIÓN.....	23
2.2 PROCESO DE DISEÑO.....	25

2.3	REQUERIMIENTOS DE DISEÑO	26
2.4	DISEÑO CONCEPTUAL	27
2.5	PARÁMETROS DISEÑO DE UN DRON	28
2.6	SELECCIÓN DE LA APLICACIÓN DEL DRON.....	28
2.7	SELECCIÓN DE COMPONENTES	29
2.7.1	Selección de motores y hélices	31
2.7.2	Potencia de vuelo en un punto fijo.....	31
2.7.3	Potencia de vuelo en ascenso.....	33
2.7.4	Potencia en descenso vertical	33
2.7.5	Potencia de motor.....	34
2.8	SELECCIÓN DE LA BATERÍA	37
CAPÍTULO 3		
3.	RESULTADOS Y ANÁLISIS.....	45
3.1	DISEÑO DE SOFTWARE	46
3.1.1	Front-end.....	46
3.1.2	Back-end	46
3.2	DISEÑO MECÁNICO.....	47
3.2.1	Brazos de dron	47
3.2.2	Base de motores.....	49
3.2.3	Dron.....	50
3.3	ANÁLISIS DE COSTOS	51
CAPÍTULO 4		
4.	CONCLUSIONES Y RECOMENDACIONES	52
	CONCLUSIONES.....	52
	RECOMENDACIONES	53
	BIBLIOGRAFÍA.....	54
	APÉNDICES.....	56

ABREVIATURAS

ESPOL	Escuela Superior Politécnica del Litoral
UAV	Unmaned Aerial Vechicle
VANT	Vehículo aéreo no tripulado
DJI	Dà-Jiāng Innovations
ESC	Electronic Speed Contoller (Controlador de velocidad electrónico)
STL	Stereolithography (estereolitografía)
CAD	Computer-Aided Design (Diseño Asistido por Computadora)
MAM	Motor a Motor

SIMBOLOGÍA

mil	milésimas de pulgada
Kg	Kilogramo
mg	Miligramo
pH	Potencial de Hidrógeno
m	Metro
mV	Milivoltio
Ni	Níquel
Mn	Manganeso
P	Fósforo
Cd	Cadmio
Li	Litio

Ángulos de Euler:

ϕ Giro alrededor del eje X, denotado como Pitch o Cabeceo.

θ Giro alrededor del eje Y, denotado como Roll o Alabeo.

ψ Giro alrededor del eje Z, denotado como Yaw o Guiñada.

Magnitudes

KV Kilo volts asignados de un motor brushless. Relaciona la geometría, número de bobinado e imanes.

A / mA Amperios / Miliamperios.

mAh Miliamperios por hora.

V Voltios.

Kg / g / mg Kilogramo / Gramo / Miligramo.

Km / m / cm / mm Kilómetro / Metro / Centímetro / Milímetro.

GHz / MHz Gigahercios / Megahercios.

GB / MB Gigabytes / Megabytes.

Mbps Megabytes por segundos.

Mbps Megabytes por segundos.

ÍNDICE DE TABLAS

Tabla 2.1 Criterios de selección por el método de los criterios ponderados.....	24
Tabla 2.2 Matriz de decisión.....	25
Tabla 2.3 Requerimiento de diseño.....	26
Tabla 2.4 Clasificación de drones por tamaño. Elaboración propia.....	29
Tabla 2.5. Ejemplos de baterías y controladores usados en drones de diferentes tamaños. Elaboración propia.....	30
Tabla 2.6. Empuje de motor 2212 a 920kv.....	35
Tabla 3.1. Análisis de costos.....	51

ÍNDICE DE FIGURAS

Ilustración 1.1 Dron DJI mini 2 tipo cuadricóptero	15
Ilustración 1.2 Motores	16
Ilustración 1.3 Hélices	16
Ilustración 1.4 Logo OpenScad	18
Ilustración 1.5 Entorno de OpenScad.....	18
Ilustración 1.6 Figura diseñada en OpenScad.....	19
Ilustración 1.7 Logo de lenguaje de programación Python	19
Ilustración 1.8 Lanzamiento de DH.82 Madre abeja (Madre de drones) “Target dron” .	20
Ilustración 1.9 Predator RQ1 dron	21
Ilustración 2.1 Flujograma del proceso de diseño del proyecto	26
Ilustración 2.2 Diseño Conceptual del Proyecto	27
Ilustración 2.3 Flujograma de proceso de diseño de un dron. Elaboración propia	28
Ilustración 2.4 Especificaciones (dimensionamiento de estator y revoluciones voltio) del motor. Elaboración propia.....	36
Ilustración 2.5 Nomenclatura y formato de hélices	37
Ilustración 2.6 Características de una batería. Elaboración propia.....	38
Ilustración 2.7 Metodología del diseño de dron	41
Ilustración 2.8 Pasos para el diseño del dron base	42
Ilustración 2.9 Sección de barra a analizar, Vista lateral del cuerpo aproximándola a una viga doblemente empotrada.	43
Ilustración 2.10. Diagrama de esfuerzo cortante	43
Ilustración 2.11 Modelo de dron generado con código. Elaboración propia	44
Ilustración 3.1. Esbozo de software encargado de diseñar un dron clase cuadricóptero: (1) Sección la cual ingresarán las especificaciones del dron tales como dimensiones, aplicaciones, controlador entre otras. (2) sección donde se visualiza el modelo de manera Automático.....	45
Ilustración 3.2. Diseño 3D del dron obtenido en formato STL	45
Ilustración 3.3 Interfaz gráfica del proyecto	46
Ilustración 3.4. código en Python, importaciones de librerías, definiciones de funciones	47

Ilustración 3.5. funciones las cuales se encargan de dibujar en OpenScad	47
Ilustración 3.6. Factor de seguridad	48
Ilustración 3.7. Análisis de fatiga por Von mises	48
Ilustración 3.8. Análisis de tensión estática	48
Ilustración 3.9 de fatiga por Von mises, se toma en cuenta solo el peso del motor.	49
Ilustración 3.10. Análisis de fuerza, considerando la fuerza generada por el motor.....	49
Ilustración 3.11 Factor de seguridad	50
Ilustración 3.12 Análisis de fatiga por Von Mises.	50
Ilustración 3.13 Análisis de tensión estática.	51

ÍNDICE DE PLANOS

PLANO 1 Vista general diseño de fuselaje con dron.....	64
PLANO 2 Cubierta superior del fuselaje.....	65
PLANO 3 Brazo del cuadricóptero.....	66

CAPÍTULO 1

1. INTRODUCCIÓN

Los drones o UAVs (Vehículos Aéreos No Tripulados) en el país se consideran equipos tecnológicos innovadores y relativamente nuevos, que surgieron como resultado de la cuarta revolución industrial, que se centra en el control, la automatización y la optimización de procesos. Actualmente, se utilizan en diferentes actividades profesionales, como el levantamiento fotogramétrico aéreo, fines publicitarios y labores de agroquímica entre otros.

Los avances tecnológicos en el campo de los drones han permitido la creación de diferentes tipos de estos dispositivos, como los drones de tipo hélicóptero y los de tipo avión. La elección de uno u otro dependerá de la aplicación en la que se utilizarán, ya que cada uno tiene sus propias ventajas y desventajas.

Los drones de tipo hélicóptero suelen ser más útiles para tareas que requieran un vuelo corto o que impliquen sobrevolar un mismo punto durante un tiempo prolongado, ya que cuentan con una mayor capacidad de maniobrabilidad y estabilidad en el aire. Por otro lado, los drones tipo avión son mejores para recorrer grandes distancias o para realizar tareas que requieran un desplazamiento rápido, ya que suelen tener una mayor velocidad y autonomía.

Desde su concepción y desarrollo, impulsados por la primera y segunda guerra mundial, su desarrollo se ha acelerado en las últimas décadas, en colaboración con tecnologías de vanguardia como: el análisis de datos, la inteligencia artificial, la electrónica y los sistemas de control.

Estos equipos suelen ser en su mayoría de un alto costo, debido a su alta demanda en el mercado para trabajos que requieran mayor producción en un menor tiempo, ya que, el presente uso de drones en el ámbito profesional posee distintas aplicaciones las cuales abarcan una gran cantidad de campos como los antes mencionados, y se pretende seguir abarcando más. No cabe duda de que los drones son una importante herramienta complementaria ya sea desde emergencias, pasando por la agricultura, ganadería, seguridad, son algunas de las áreas donde los drones han logrado facilitar y optimizar el trabajo [3].

1.1 Descripción del problema

En la actualidad, la innovación tecnológica ha llevado consigo cambios significativos en la industria ecuatoriana, y uno de estos cambios es la introducción de drones en una amplia variedad de sectores profesionales [3]. Estos dispositivos han demostrado ser una herramienta valiosa al aumentar la eficiencia y productividad en un menor tiempo de trabajo. Sin embargo, la adquisición de drones de alta calidad a menudo se ve obstaculizada por el reglamento de uso de drones y un elevado costo, el cual está relacionado con la importación, los aranceles e impuestos [4].

Ecuador se encuentra en una etapa incipiente en cuanto a la producción y diseño de drones. A pesar de que algunas empresas se han especializado en la fabricación de esta tecnología [5]. La mayoría todavía prefiere importarlos para utilizarlos en aplicaciones como la fotografía, investigación, el levantamiento topográfico y la agricultura entre otras. Por lo tanto, es imperativo realizar un estudio y desarrollo de herramientas para optimizar la fabricación de drones multipropósito VANT. Con ello, se espera impulsar el sector de la producción de drones en el país, al mismo tiempo que se brinda una solución más eficiente y efectiva a las necesidades actuales.

En este proyecto, nos enfocamos en el desarrollo de un software que permita el diseño de drones específicos para trabajar en áreas de minería y agricultura. La optimización de este proceso permitirá reducir los costos asociados a la mano de obra y la importación de productos con un elevado valor monetario, al mismo tiempo que mejorará la eficiencia y productividad en estos sectores. [3] Este proyecto se enfoca en el desarrollo de un código apto para el diseño de drones, capaces de realizar trabajos en áreas de minería y agricultura, ya que el Ecuador se encuentra en una etapa inicial respecto a la producción y diseño de este producto. A pesar de que existen ciertas empresas dedicadas a la manufactura de esta tecnología, se sigue prefiriendo la importación de esta con la finalidad de llevar a cabo trabajos de fotografía, levantamiento topográfico, agricultura en la aplicación de agroquímicos, entre otros. por consiguiente, se necesita realizar el respectivo estudio y desarrollo de herramientas las cuales sean óptimas para el diseño y fabricación de drones multipropósitos VANT

1.2 Justificación del problema

El aumento de la demanda en el mercado local e internacional ha llevado a las empresas a buscar formas de mejorar la producción en sus áreas de trabajo. Muchas de ellas han optado por aumentar la mano de obra o invertir en tecnologías para lograr este objetivo, pero estas soluciones no han resultado ser suficientes. La mano de obra extra a menudo está sujeta a periodos de inactividad y la tecnología puede ser muy costosa.

Es aquí donde entra en juego este proyecto innovador. La idea es desarrollar y optimizar el proceso de diseño de un dron a través de un software basado en scripts que se adapte a las especificaciones del trabajo que debe realizar el dron. De esta manera, se busca reducir los costos de mano de obra y la importación de productos con un alto valor monetario.

El uso de drones en la industria ha demostrado ser una herramienta valiosa en la automatización de procesos y mejora de la eficiencia. Sin embargo, muchas empresas aún no han adoptado esta tecnología debido a los altos costos y la falta de personalización en el diseño. Con este proyecto, se busca solucionar estos problemas y brindar una solución asequible y efectiva a las empresas interesadas en mejorar su producción y convertirse en líderes en la innovación.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar un software para el dimensionamiento, optimización y diseño de un vehículo aéreo no tripulado (VANT) clase cuadricóptero, basado en OpenScad aplicando conocimientos en programación.

1.3.2 Objetivos Específicos

- Construir un algoritmo para el dimensionamiento de vehículos aéreos no tripulados mediante el uso de ecuaciones basadas en la teoría de construcción de drones.
- Programar funciones de generación automática para el modelado 3D de partes del fuselaje de un dron mediante la programación funcional usando un software CAD de código abierto.
- Validar un prototipo virtual mediante el modelado 3D y el ensamblaje correcto de las piezas, visualizando el producto es otros softwares de diseño 3D

1.4 Marco teórico

1.4.1 Componentes de un dron

Los componentes de un dron son los elementos esenciales que le permiten funcionar y conformar su estructura. Estos componentes incluyen el marco o fuselaje, los motores, las hélices, la batería, el controlador de vuelo, la cámara, los sensores y el sistema de transmisión de datos. La selección y diseño de cada componente es crucial para garantizar el correcto funcionamiento del dron, su durabilidad, resistencia y eficiencia. Además, la utilización de técnicas de fabricación avanzadas, como la impresión 3D y el corte láser, permite optimizar la producción de algunos componentes y mejorar su rendimiento. En general, la fabricación de drones requiere un enfoque en la optimización de los materiales para lograr un equilibrio óptimo entre resistencia y peso ligero [6].

Chasis: El chasis es una parte fundamental de un dron, ya que es donde se ubican todos los componentes que permiten al dispositivo volar y realizar sus funciones. La forma del chasis dependerá del número de rotores que tenga el dron, ya que cada uno de ellos se encarga de impulsar el vehículo en el aire.

Si el chasis es muy grande, el peso del dron aumentará y se necesitarán motores o hélices más potentes para sostenerlo en el aire. Esto puede influir en la duración de la batería y en la carga útil que el dron pueda llevar, limitando así su capacidad de desempeño. Por eso, es importante elegir un chasis adecuado para cada tipo de dron, teniendo en cuenta sus características y necesidades.

Según el número de rotores que tenga un dron, se le puede denominar de diferentes maneras. Por ejemplo, en la Ilustración 1.1 se muestra un dron con cuatro rotores por lo que se llamaría cuadricóptero, mientras que uno con seis se llama hexacóptero y uno con ocho se llama octacóptero. Cada uno de estos diseños tiene sus propias ventajas y desventajas, y se utilizan para diferentes aplicaciones.



Ilustración 1.1 Dron DJI mini 2 tipo cuadricóptero [7]

Motores: elementos capaces de mantener en el aire el dron, girando las hélices, por lo general los motores más usados, son motores eléctricos, que funciona mediante una bobina en la cual circulará corriente eléctrica como se puede observar en la Ilustración 1.2. Los cuales pueden ser: [8]

- Trifásicos, de mayor potencia y precisión, gran rango de velocidad de giro, con una menor interferencia electromagnética.
- bifásicos, hechos de un material conductor encargado de transmitir la intensidad hacia la bobina, logrando crear un campo magnéticos, por lo regular, suelen ser más pesados que los motores Trifásicos, con un menor tiempo de vida útil.



Ilustración 1.2 Motores [9]

Hélices: Encargadas junto con el motor de elevar el dron en el aire, ya que son giradas por la potencia que es transmitida por los motores. Estas pueden ser de distintas aspas, dependiendo que se priorice como la estabilidad, el consumo de energía, entre otros factores. Asimismo, entre mayor longitud tenga las hélices, mayor será su empuje, de igual forma su consumo de corriente [8]. En pocas palabras, son aquellos elementos del dron que permiten crear la fuerza de empuje en vertical necesario para sus movimientos. Como se puede observar en la Ilustración 1.3 ejemplos de aspas



Ilustración 1.3 Hélices [6]

Baterías: Poseen la capacidad de suministrar energía al dron, en otras palabras, vendría a ser el corazón del UAV, en su mayoría suelen ser recargables. De igual forma, se pueden encontrar distintos tipos de baterías para drones.

- Níquel – cadmio (Ni-Cd) baterías de níquel-cadmio, no son de carga rápida, además, poseen efecto memoria [8].
- Níquel -Mercurio e Hidrogeno (Ni-MH) baterías de níquel-metal-hidruro. Tienen una menor vida útil, poseen efecto memoria [8].
- Ion-Litio baterías ligeras, sin efecto memoria, con una menor descarga cuando están almacenadas, sin embargo, en caso de sufrir algún daño o perforación y al reaccionar sus componentes con oxígeno estas resultarían peligrosas, ya que, son inflamables [8].
- Li-Po Actualmente las baterías más modernas en el mercado, optimiza el espacio de fuselaje debido a su fabricación, no poseen efecto memoria. Sin embargo, son inflamables al igual que las anteriores [8].

Placa controladora de vuelo: Considerado también como el cerebro del dron, debido a que es el ordenador integrado que recoge datos del sistema del dron, GPS, velocidades, información de giroscopio, acelerómetros.

La placa de vuelo puede tener incorporada un autopiloto que le permite al dron poder mantenerse en el aire de forma autónoma sin necesidad de un operador. Aquí también se procesa la información de transmisión y recepción de datos, esto sirve de nexo entre el operador y el vehículo; los canales que requiere el dron para funcionar dependen del número de motores que tenga.

La función de emisión de datos es propia del control remoto que posee el operador, cuando el control emite una acción, esta señal es captada por el receptor que se encuentra en el dron.

1.4.2 Programación OpenScad

OpenScad: es un software CAD como se muestra en la Ilustración 1.4, de código abierto dirigido a crear modelos sólidos en 3D, mediante scripts el cual utiliza su propio lenguaje de descripción, por consiguiente, no se puede interactuar de manera directa con el modelo previsualizado. Los modelos se crean utilizando una técnica llamada geometría sólida constructiva. Según esta técnica, los objetos simples se pueden transformar y combinar para crear casi cualquier modelo complejo.

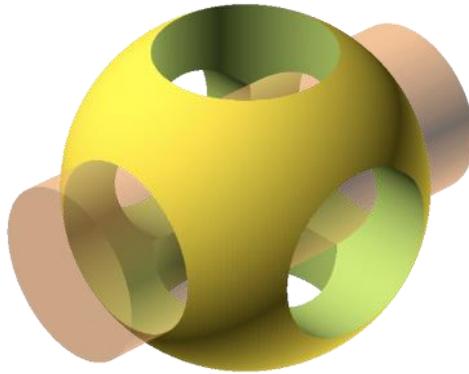


Ilustración 1.4 Logo OpenScad

Entorno de trabajo OpenScad: se encontrará una ventana la cual está dividida en dos secciones. La sección de la izquierda es un editor de texto, donde se escribirá el lenguaje con el que se maneja OpenScad; mientras que la parte derecha es el espacio 3D virtual donde se previsualizará los modelos realizados en la sección izquierda, observándose lo antes mencionado en la Ilustración 1.5

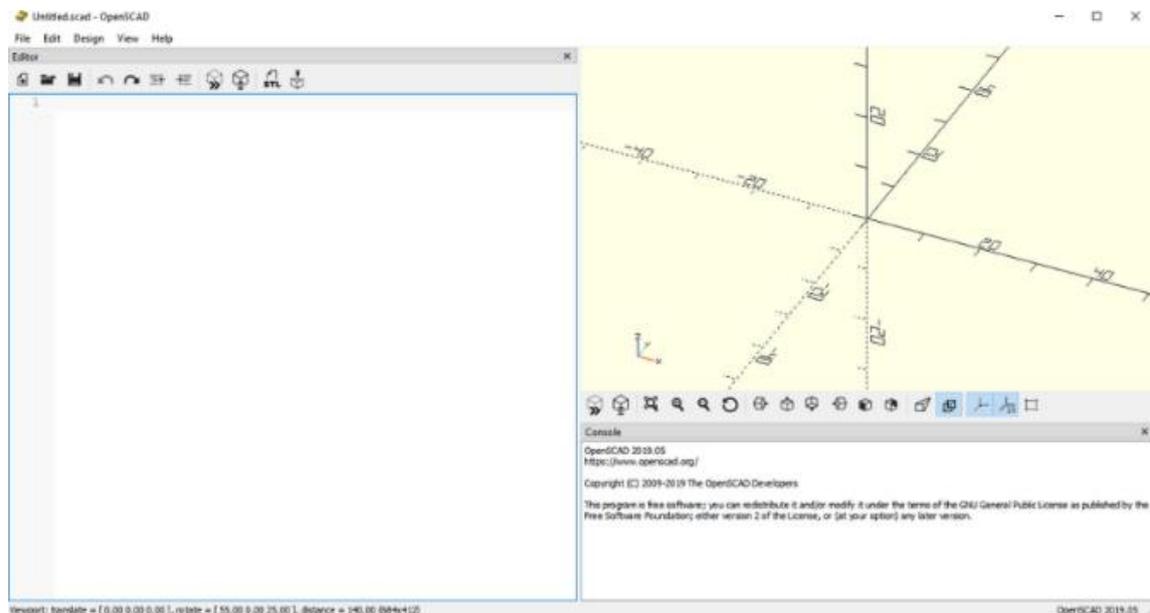


Ilustración 1.5 Entorno de OpenScad

La programación en OpenScad se basa en creación de figuras 3D básicas como: cubos, prismas, cilindros, esferas, conos. Además, permite hacer operaciones con estas figuras en 3D complejas, desde realizar operaciones booleanas de unión e intersección, a usar operaciones de rotación, translación y escalamiento. El conjunto de estas instrucciones puede dar como resultado piezas con cierto grado de complejidad, en el

ejemplo de la Ilustración 1.5 se puede observar que el código genera un boceto simple de un coche.

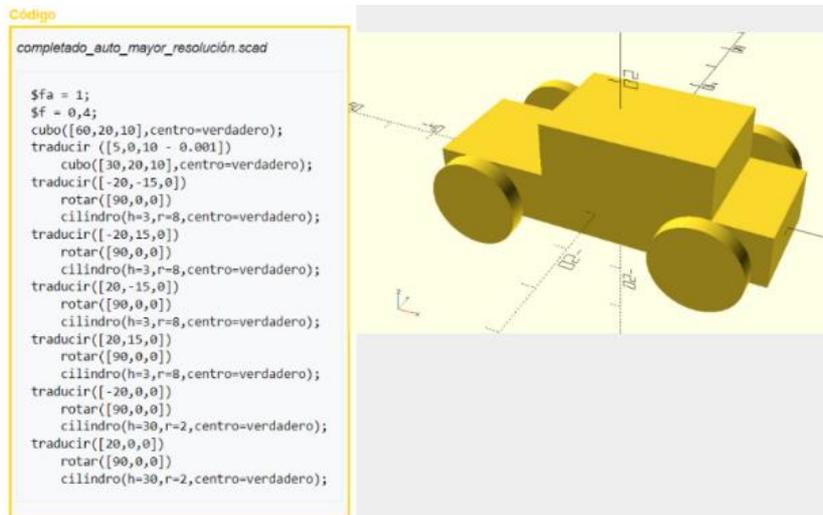


Ilustración 1.6 Figura diseñada en OpenScad

1.4.3 Lenguaje de programación

Es un conjunto de instrucciones mediante el cual un usuario (humano) interactúa con las computadoras. Debido a que se permite comunicarse mediante una serie de algoritmos, e instrucciones escritas en una sintaxis que la computadora se encarga de interpretar en lenguaje de máquina [10].

Python Ilustración 1.7: es un lenguaje de programación multiplataforma de código abierto, el cual es muy versátil ya que se puede usar tanto para desarrollo web, creación de software, procesamiento de datos entre otros. Además, que es un lenguaje muy versátil debido a sus grandes cantidades de librerías las cuales son usadas para distintas operaciones [11].



Ilustración 1.7 Logo de lenguaje de programación Python [11]

Librerías en python: Corresponden al conjunto de implementaciones que permiten codificar este lenguaje, con el objeto de crear una interfaz independiente. De las cuales se destacará **Openpyscad** la cual permite manejar datos 3D del programa Openscad; **win32gui y win32con** conjunto de implementaciones capaces de correr un código o una interfaz gráfica en el sistema operativo Windows.

1.4.4 Estado del arte

Esto debido que, desde su creación, los drones han experimentado una evolución constante en términos de diseño y tecnología. Los drones o VANT fueron de las primeras aeronaves no tripuladas siendo diseñada y construida a lo largo de la primera guerra mundial en el año 1916 como se puede apreciar en la ilustración 1.7 estas aeronaves eran denominadas “Air target” (Ilustración 1.8) el cual se podía controlar mediante radiofrecuencia AM [12].



Ilustración 1.8 Lanzamiento de DH.82 Madre abeja (Madre de drones) “Target dron” [1].

Sin embargo, con el tiempo, su uso se ha expandido a una amplia gama de aplicaciones, incluyendo la vigilancia, la fotografía aérea y el entretenimiento. En esta revisión del estado del arte, se analizarán los desarrollos más importantes en el diseño de los drones, desde sus orígenes hasta la actualidad.

Los drones surgieron en la década de 1950, cuando la aviación militar comenzó a utilizarlos para fines de reconocimiento y vigilancia. En 1994, se produjo un hito importante en la evolución de los drones con el lanzamiento del

primer dron que incorporaba GPS, el RQ-Predator (Ilustración 1.9). Este avance permitió una mayor precisión en la navegación y control, y sentó las bases para el desarrollo de modelos cada vez más avanzados [13].



Ilustración 1.9 Predator RQ1 dron [13]

En la década de 2000, los drones comenzaron a ser utilizados con más frecuencia en aplicaciones civiles, como la fotografía aérea y la vigilancia. Esto se debió en parte a la disponibilidad de tecnologías de vuelo más avanzadas y a la reducción de los costos de producción. Al mismo tiempo, los fabricantes de drones comenzaron a desarrollar modelos más pequeños y maniobrables, lo que permitió su uso en una amplia gama de entornos.

En la actualidad, los drones están disponibles en una amplia gama de tamaños, características y precios. Los modelos más avanzados cuentan con cámaras de alta resolución, sistemas de navegación autónoma, sensores de posicionamiento y tecnología de inteligencia artificial, lo que les permite realizar tareas complejas con una gran precisión.

Además, los fabricantes de drones están desarrollando constantemente nuevas tecnologías y mejoras en el diseño, lo que les permite ser más compactos, ligeros y eficientes. Estos avances también han permitido la integración de nuevas funciones, como la capacidad de transmitir imágenes en tiempo real y la compatibilidad con aplicaciones móviles para el control remoto [14].

A pesar de los avances en el diseño y la tecnología de los drones, existen todavía algunos desafíos importantes que deben ser abordados. Uno de los mayores desafíos es la seguridad y el control de los drones en el espacio aéreo, especialmente en áreas urbanas y cerca de aeropuertos. Para abordar este problema, los reguladores gubernamentales están desarrollando regulaciones y normas para regular el uso de drones en el espacio aéreo [4].

Otro desafío importante es la duración de la batería y la autonomía de los drones. Aunque los avances tecnológicos han mejorado significativamente la eficiencia energética, todavía hay una limitación en la duración de la batería. Los fabricantes están trabajando en soluciones para mejorar la duración de la batería, como la utilización de baterías de mayor capacidad y la optimización de los sistemas de control y navegación.

En resumen, los drones han experimentado una evolución constante en términos de diseño y tecnología desde sus orígenes como una herramienta militar hasta su uso en una amplia gama de aplicaciones civiles y profesionales. Con la continuación de los avances en tecnología y diseño, se espera que los drones sigan desempeñando un papel importante en la sociedad y en la economía en el futuro.

CAPÍTULO 2

2. METODOLOGÍA

En este capítulo se presenta un análisis de las distintas alternativas de solución para la selección de software en el cual se desarrollará el proyecto, flujograma que detalla los pasos a seguir para el cumplimiento del proyecto y el diseño respectivo del sistema.

2.1 Selección de la alternativa de solución

Partiendo de la problemática descrita en el capítulo anterior, se definieron 3 alternativas de solución para el desarrollo de un software de diseño de modelado y dimensionamiento de un dron.

- **Alternativa 1:**

Desarrollar un software mediante el lenguaje de programación Python, junto con las respectivas librerías que nos permitan conectar con la aplicación de software libre OpenScad. Para el dimensionamiento, optimización y diseño de un VANT clase cuadricóptero.

- **Alternativa 2:**

Diseñar VANT clase cuadricóptero mediante el software CAD Inventor AUTODESK, usando las respectivas herramientas de la aplicación como ifeatures, entre otras.

- **Alternativa 3:**

Desarrollar un software mediante el lenguaje de programación Python el cual permite modificar con la aplicación SolidWorks para el respectivo diseño de un dron multipropósito.

Para seleccionar la mejor alternativa, se efectuó un análisis de cada una de ellas en función de los criterios establecidos en donde por medio del método de los criterios ponderados como se ilustra en la Tabla 2.1 en la que se indican los rangos de importancia, en otras palabras, los criterios que predominan y su rango de importancia según los establecida

- **Preferencia del cliente:** de las alternativas antes mostradas, por cual el cliente desea optar para desarrollar el proyecto.
- **Costo:** el precio de software a usar, junto a las herramientas que conlleva para el diseño de un dron.
- **Facilidad de uso:** Software fácil de usar, seria más atractivo para usuarios con diferentes niveles de experiencia en software CAD.
- **Fiabilidad:** Un software confiable es esencial para garantizar que el proceso de diseño sea consistente y preciso.
- **Flexibilidad:** Capacidad de adaptarse a las diversas circunstancias a las distintas necesidades.
- **Escalabilidad:** capacidad de ampliación del sistema para satisfacer las necesidades de diseño.

Peso de Criterios	Flexibilidad > Preferencia del Cliente = Facilidad de Uso = Escalabilidad > Fiabilidad > Costo							
Criterio	Preferencia del Cliente	Costo	Facilidad de Uso	Fiabilidad	Flexibilidad	Escalabilidad	∑+1	Ponderación
Preferencia del Cliente	1	0,5	0	0,5	0,5	0	2,5	0,18
Costo	0	1	0,5	0	0	0	1,5	0,11
Facilidad de Uso	0,5	0	1	0	1	0	2,5	0,18
Fiabilidad	0,5	0	0,5	1	0	0	2	0,14
Flexibilidad	0	0	1	0,5	1	0,5	3	0,21
Escalabilidad	0	0	0,5	0,5	0,5	1	2,5	0,18
							14	1,00

Tabla 2.1 Criterios de selección por el método de los criterios ponderados

Según la

Peso de Criterios	Flexibilidad > Preferencia del Cliente = Facilidad de Uso = Escalabilidad > Fiabilidad > Costo							
Criterio	Preferencia del Cliente	Costo	Facilidad de Uso	Fiabilidad	Flexibilidad	Escalabilidad	∑+1	Ponderación
Preferencia del Cliente	1	0,5	0	0,5	0,5	0	2,5	0,18
Costo	0	1	0,5	0	0	0	1,5	0,11
Facilidad de Uso	0,5	0	1	0	1	0	2,5	0,18
Fiabilidad	0,5	0	0,5	1	0	0	2	0,14
Flexibilidad	0	0	1	0,5	1	0,5	3	0,21

Escalabilidad	0	0	0,5	0,5	0,5	1	2,5	0,18
							14	1,00

Tabla 2.1 la prioridad de los criterios de son **Flexibilidad > preferencia = facilidad de uso = escalabilidad > fiabilidad > costo**. Por consiguiente, en el APÉNDICE A se desglosa más a detalle sobre estos datos, en donde se evaluó cada alternativa para cada criterio obteniendo así la matriz de decisión Tabla 2.2 en la que se puede analizar que la alternativa es considerado mejor en comparación a la alternativa 2 y 3. Por consiguiente, esta alternativa es la óptima para el desarrollo del proyecto.

Tabla 2.2 Matriz de decisión

Pesos	Criterios de Selección						Resultados	
	0,18	0,11	0,18	0,14	0,21	0,18	$\Sigma+1$	Prioridad
Conclusión	Preferencia del cliente	Costo	Facilidad de uso	Fiabilidad	Flexibilidad	Escalabilidad		
Alternativa 1	0,4	0,4	0,34	0,3	0,4	0,4	0,38	1
Alternativa 2	0,3	0,3	0,43	0,4	0,3	0,2	0,32	2
Alternativa 3	0,2	0,3	0,25	0,3	0,3	0,4	0,29	3
							1,0	

2.2 Proceso de diseño

Con el objetivo de alcanzar el diseño se presenta un flujograma en la Ilustración 2.1, donde fue importante recorrer por varias etapas, abordando distintas áreas como fue el conocer el problema o necesidad, proponer las alternativas de solución en conjunto con la ponderación de estas y elegir una solución de las alternativas propuestas para posteriormente realizar modelos, análisis y selección de elementos para el correcto desarrollo del proyecto

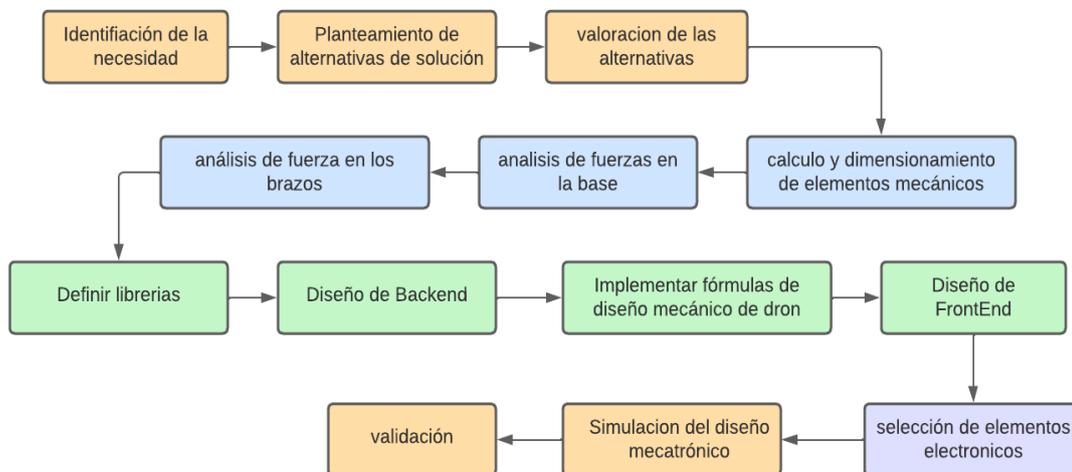


Ilustración 2.1 Flujograma del proceso de diseño del proyecto

2.3 Requerimientos de diseño

Para realizar el desarrollo del software de diseño de un dron multipropósito clase cuadricóptero, fue importante conocer con anticipación los requerimientos del cliente quien dio a conocer con los requisitos. Por ende, en la Tabla 2.3 se detallan los requerimientos establecidos.

Tabla 2.3 Requerimiento de diseño

Funcionalidad	Diseñar un modelado 3D de un dron de forma autónoma, mediante el ingreso de parámetros de entrada.
Modelado y dimensionamiento	Ser capaz de realizar el delado y dimensionamiento del dron de manera confiable, tomando en cuenta su aplicación, análisis estructural, componentes y más.
Compatibilidad	Compatible con el sistema operativo Windows y la aplicación OpenScad.
Herramientas	Se debe usar la aplicación OpenScad y Python.
Tamaño	Dimensiones máximas del dron deben ser inferiores a 1000 x 1000 mm, para que pueda ser impreso fácilmente en 3D.
Librerías	Debe usar librerías como: math, openpyscad, threading, PySimpleGUIQt, win32gui, win32con.

Costos

Debe costar menos a comparación del costo actual de la contratación de un diseñador mecánico y la suscripción de un programa CAD de diseño 3D.

2.4 Diseño conceptual

En la Ilustración 2.2 se muestra un boceto inicial sobre el diseño del software de dimensionamiento y modelado de un dron cuadricóptero. En ella podemos identificar y seleccionar, componentes tales como el tamaño del dron, los componentes (placa de vuelo, motores, cámara, batería), la carga, la duración de vuelo, y el dimensionamiento del producto (dron). En este caso se realizará los cálculos serán realizados para un dron determinado, con cierto dimensionamiento, potencia de vuelo, potencia de motores entre otros aspectos.

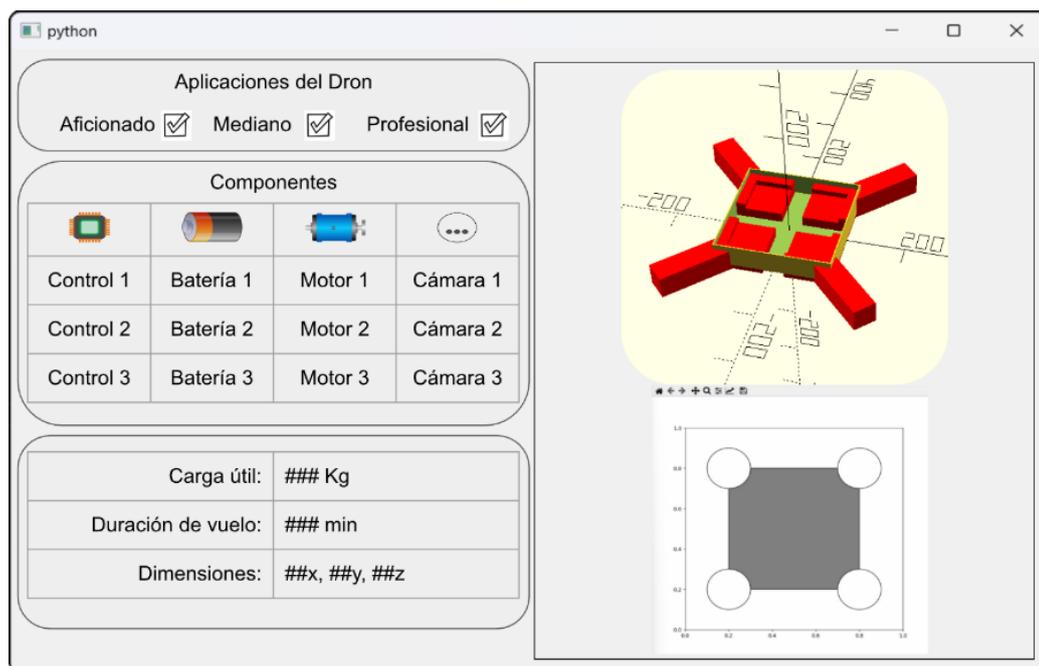


Ilustración 2.2 Diseño Conceptual del Proyecto

Para el desarrollo del software fue necesario cumplir con los requerimientos de diseño, abordando principalmente la funcionalidad de generar un dron de forma autónoma a partir de parámetros de entrada. Para lograr este propósito, se utiliza el lenguaje de programación Python, ya que, al ser un lenguaje de alto nivel, cuenta con librerías para el diseño 3D y la implementación de una interfaz gráfica intuitiva para el

usuario. Dado que Python es un lenguaje Open Source nativo de Linux, la aplicación podrá ser ejecutada en otros sistemas operativos basados en Debian como Ubuntu o en Raspberry Os.

2.5 Parámetros diseño de un dron

Para iniciar con el diseño del dron, es fundamental obtener los parámetros a utilizar los cuales serán extraídos por medio de la interfaz gráfica. Para ejemplificar la metodología mostrada en el flujograma de la Ilustración 2.3 seguirá el software, se analizará un caso de aplicación para un dron cuadricóptero usado en investigación. A continuación, se describirán detalladamente cada uno de estos pasos.

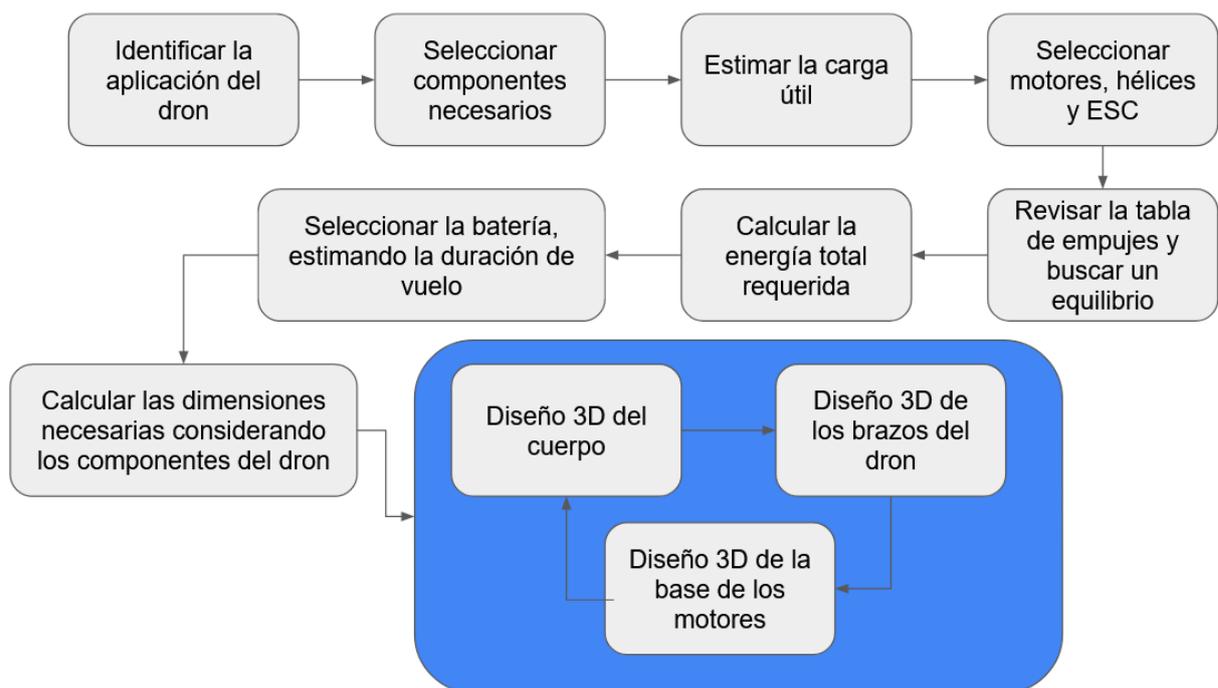


Ilustración 2.3 Flujograma de proceso de diseño de un dron. Elaboración propia

2.6 Selección de la aplicación del dron

En esta etapa es importante determinar la aplicación y en qué entorno se desempeñará la aeronave, esto decidirá sus dimensiones generales y otras características importantes del diseño.

Existen varias aplicaciones para los drones, y cada una de ellas puede ser más adecuada para un tamaño de dron específico. En la Tabla 2.4 se ven algunos ejemplos, los drones pequeños con un diámetro de motor a motor (MAM) de 215 milímetros similares al MAVIC PRO suelen ser más ágiles y maniobrables, lo que los hace ideales para aplicaciones de vídeo y fotografía aérea, mientras que los drones más grandes como el PHANTHOM 5 pueden llevar cargas pesadas y tienen un mayor alcance, lo que los hace adecuados para aplicaciones de transporte y vigilancia.

En nuestro caso se necesita un dron de bajo costes que sirva de ejemplo práctico para el funcionamiento de nuestro software, para ello se seleccionó un dron de un tamaño mediano, de esta forma puede servir tanto para investigación como para ciertas aplicaciones profesionales.

Tamaño	Aplicaciones	Dimensiones MAM	Ejemplos	Dimensiones
pequeño	civil, aficionado, infantiles, investigación.	130 a 230 mm	MAVIC PRO 	198 x 83 x 83 mm (lengthxwidthxheight) aprox 215 mm (MAM)
mediano	civil, profesional, militar, fotogrametría, construcción, minería	230 a 330 mm	MAVIC PRO 2 	214 x 91 x 84 mm (lengthxwidthxheight) aprox 233 mm (MAM)
Grande	profesional, militar, fotogrametría, construcción, minería, agricultura.	330 a 600 mm	PHANTHOM 4 	289 x 289 x 196 mm (lengthxwidthxheight) aprox 415 mm (MAM)

Tabla 2.4 Clasificación de drones por tamaño. Elaboración propia

2.7 Selección de componentes

En general, la mayoría de los drones incluyen componentes comunes como baterías, tarjetas de control de vuelo, controladores de velocidad y motores. Cada uno de estos componentes tiene un peso y unas dimensiones, siendo el peso el factor más importante a tener en cuenta.

En la Ecuación 2.1 se muestra cómo se calcula W_{dron} (peso total del dron) teniendo en consideración el peso del fuselaje y de sus componentes electrónicos.

$$W_{dron} = W_{batería} + W_{controlador} + W_{motores} + W_{fuselaje} \quad 2.1$$

En la Tabla 2.5 se muestra ejemplos de pesos y dimensiones de algunos componentes usados en drones de diferentes tamaños, nuestro dron será de tamaño mediano y se considera que usará una batería LiPo de 3 celdas con 2200 mAh y un controlador de vuelo ligero como el SP Racing F3.

	Batería			Controladores		
	descripción	peso	dimensiones	descripción	peso	dimensiones
Dron Pequeño	2S (7,4 V y 1500 mAh)	110 gramos	105 x 35 x 18 mm	SP Racing F3	15 gramos	36 x 36 x 6 mm
Dron Pequeño	2S (7,4 V y 2500 mAh)	170 gramos	123 x 36 x 34 mm	F4 Omnibus	20 gramos	36 x 36 x 8 mm
Dron Mediano	3S (11,1 V y 1100 mAh)	100 gramos	102 x 34 x 16 mm	APM 2.8	25 gramos	50 x 50 x 15 mm
Dron Mediano	3S (11,1 V y 2200 mAh)	150 gramos	135 x 42 x 25 mm	Holybro Kakute F7	30 gramos	45 x 45 x 8 mm
Dron Grande	4S (14,8 V y 1500 mAh)	140 gramos	105 x 35 x 22 mm	DJI N3	50 gramos	45 x 45 x 10 mm
Dron Grande	4S (14.8 V y 5000 mAh)	600 gramos	150 x 45 x 25 mm	Pixhawk 4	65 gramos	60 x 60 x 15 mm
Opcional	Bateria 18650 1S 3,7 V 2000 - 3000 mAh	45 - 60 gramos	18 x 65 mm	Arduino uno	25 gramos	68 x 53 x 15 mm

Tabla 2.5. Ejemplos de baterías y controladores usados en drones de diferentes tamaños. Elaboración propia

Tomando en consideración las características generales de un dron mediano y los componentes seleccionados se estimó un peso aproximado de 580 gr detallado en la Tabla 2.6.

Tabla 2.6 Estimación de pesos para el diseño del dron

Componente	Peso/unidad	Cantidad	Total
Motores 2212 920kv	50 gr	4	200 gr
Controladora SP Racing	20 gr	1	20 gr
ESC	25 gr	4	100 gr
Batería 3S	150 gr	1	150 gr
Cables	10 gr	1	10 gr
Hélices	25 gr	4	100 gr
Total			580 gr

2.7.1 Selección de motores y hélices

Para asegurarnos de que los motores darán el empuje necesario para mantener al dron en vuelo, primero hay que calcular la potencia requerida por cada motor; se utilizó la ecuación 2.2 que divide el peso máximo de nuestro dron para el número de rotores.

$$T = \frac{Carga_{total}}{4} \quad 2.2$$

$$T = \frac{580 \text{ gr}}{4}$$

$$T = 145 \text{ gr} \rightarrow 1.4 \text{ N}$$

Luego se determinó la potencia requerida para los diferentes tipos de vuelos que son:

2.7.2 Potencia de vuelo en un punto fijo

La potencia de vuelo en un punto fijo es la energía necesaria para mantener al dron suspendido en el aire sin que este realice ninguna maniobra; es necesario calcular

primero el área de proyección de las hélices y la velocidad inducida usando las ecuaciones 2.3 y 2.4 respectivamente.

$$A = \pi * r^2 \quad 2.3$$

$$V_{io} = \sqrt{\frac{T}{2\rho A}} \quad 2.4$$

Se consideró un conjunto de 4 hélices de 9.5 pulgadas de diámetro y una densidad del aire de 1.225 kg/m³; reemplazando en las ecuaciones 2.3 y 2.4 obtenemos que el área proyectada de las hélices corresponden a 4.5*10⁻² con una velocidad inducida de 3.56 m/s:

$$A = \pi * r^2$$

$$A = \pi * 12^2$$

$$A = 452.37 \text{ cm}^2$$

$$A = 4.5 * 10^{-2} \text{ m}^2$$

$$V_{io} = \sqrt{\frac{T}{2\rho A}}$$

$$V_{io} = \sqrt{\frac{1.4}{2 * 1.23 * 4.5 * 10^{-2}}}$$

$$V_{io} = 3.56 \text{ m/s}$$

Finalmente se calcula la potencia empleada usando la ecuación 2.5.

$$P_{io} = TV_{io} \quad 2.5$$

$$P_{io} = 1.4 * 3.56$$

$$P_{io} = 4.98 \text{ W}$$

2.7.3 Potencia de vuelo en ascenso

La potencia de vuelo en ascenso es la energía requerida para que el dron pueda elevarse; en la maniobra de ascenso se definió una velocidad de 3 m/s, gracias a la ecuación 2.6 se determinó que la velocidad inducida es de 2.36 m/s y con la ecuación 2.7 se obtuvo una potencia de 7.5 W

$$\frac{V_i}{V_{io}} = -\frac{1}{2} \left(\frac{V_c}{V_{io}} \right) + \sqrt{\frac{1}{4} \left(\frac{V_c}{V_{io}} \right)^2 + 1} \quad 2.6$$

$$\frac{V_i}{3.56} = -\frac{1}{2} \left(\frac{3}{3.56} \right) + \sqrt{\frac{1}{4} \left(\frac{3}{3.56} \right)^2 + 1}$$

$$V_i = 2.36 \text{ m/s}$$

$$P_i = T * (V_c + V_i) \quad 2.7$$

$$P_i = 1.4 * (3 + 2.36)$$

$$P_i = 7.5 \text{ W}$$

2.7.4 Potencia en descenso vertical

En el descenso vertical existe una interacción entre el flujo del área ascendente y las aspas del rotor, esta interacción genera anillos de vórtices que bordea el disco generando vibraciones; teniendo en cuenta una velocidad de descenso de 3 m/s, se usa la ecuación 2.8 para calcular una relación que debe estar entre los siguientes casos:

- Normal $V_c/V_{io} \geq 0$
- Anillos de vórtice $-1 \leq V_c/V_{io} < 0$
- Estela turbulenta $-2 \leq V_c/V_{io} < -1$
- Molinete frenante $V_c/V_{io} < -2$

$$\frac{V_c}{V_{io}} \quad 2.8$$

$$\frac{V_c}{V_{io}} = \frac{-3}{3.56}$$

$$\frac{V_c}{V_{io}} = -0.84$$

El resultado muestra la generación de vórtice, se procede a realizar una corrección con un valor de k de 1.15 usando la ecuación 2.9

$$\frac{V_i}{V_{io}} = k - \frac{V_c}{V_{io}} \quad 2.9$$

$$\frac{V_i}{V_{io}} = 1.15 - (-0.84)$$

$$V_i = V_{io}(1.15 + 0.84)$$

$$V_i = 3.56(1.15 + 0.84)$$

$$V_i = 7.08 \text{ m/s}$$

Luego se calcula su potencia en descenso con la siguiente ecuación 2.10

$$P_{id} = 1.4 * (-3 + 7.08) \quad 2.10$$

$$P_{id} = 5.71 \text{ W}$$

2.7.5 Potencia de motor

Calculado la potencia de los motores en sus diferentes modos de vuelo, es necesario aplicar un factor de corrección de 0.8 y un factor de servicio de 1.2 para asegurarnos que el dron pueda ser capaz de cumplir su propósito; en este análisis se procedió a usar la ecuación 2.11 para obtener la potencia requerida tomando de referencia la potencia de ascenso como potencia máxima y finalmente con 2.12^[OBJ] calcular la potencia de nuestro motor :

$$P_n = \frac{P_{max}}{FC} \quad 2.11$$

$$P_n = \frac{7.5}{0.8}$$

$$P_n = 9.38 \text{ W}$$

$$P_{motor} = f_s * P_n \quad 2.12$$

$$P_{motor} = 1.2 * 9.38$$

$$P_{motor} = 11.256 \text{ W}$$

Cada motor demandará una potencia igual o mayor a 11.256 W para que sea capaz de suministrar la energía necesaria y generar un empuje de 145 gr; previamente se eligió el motor 2212 920kv mostrado en la figura 2.4 y según sus especificaciones mostradas en la tabla #, es capaz de levantar 230 gr con una potencia de 20 W al 50% de su rendimiento.

Tabla 2.7. Empuje de motor 2212 a 920kv

ML2212 MOTOR								
Item NO.	Volts (V)	Prop	Throttle	Amps (A)	Watts (W)	Thrust (g)	Efficiency (g/W)	Operating temperature (°C)
ML2212 920KV	11.1V	DJI9.4*4.3	50%	1.8	20.0	230	11.5	37°C
			65%	2.8	31.1	310	10.0	
			75%	3.9	43.3	410	9.5	
			85%	5.5	61.1	480	7.9	
			100%	7.6	84.4	610	7.2	
		APC10*4.5	50%	2.6	28.9	290	10.0	55°C
			65%	5.1	56.6	460	8.1	
			75%	7.4	82.1	590	7.2	
			85%	10.1	112.1	730	6.5	
			100%	13.4	148.7	860	5.8	
	14.8V	DJI9.4*4.3	50%	2.7	40.0	350	8.8	52°C
			65%	4.4	65.1	490	7.5	
			75%	6.3	93.2	640	6.9	
			85%	8.3	122.8	790	6.4	
			100%	11.5	170.2	990	5.8	

Notes: The test condition of temperature is motor surface temperature in 100% throttle while the motor run 10 min. environment temperature 24°C

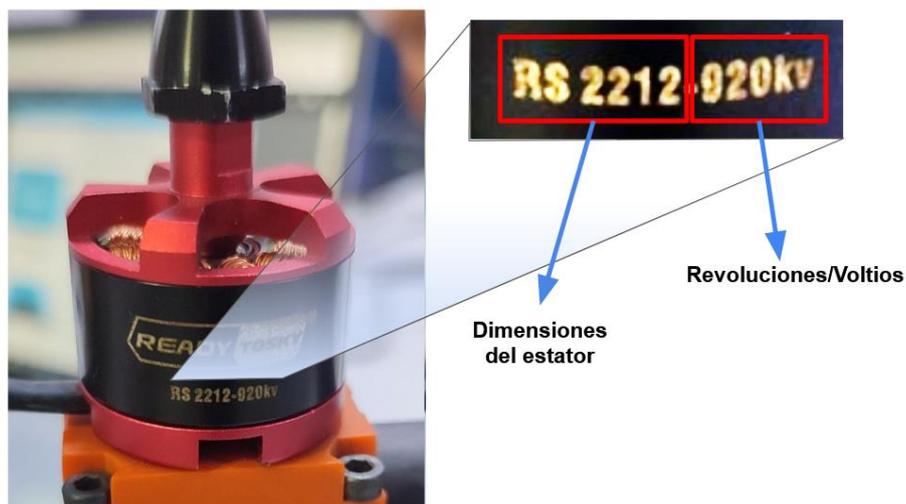


Ilustración 2.4 Especificaciones (dimensionamiento de estator y revoluciones voltio) del motor. Elaboración propia.

La nomenclatura del motor de la figura 2.4 se aprecia un código que se divide en dos números, el primero es 2212 cuyos dos primeros dígitos corresponden al diámetro en milímetros del rotor y los dígitos siguientes representan a su altura; el segundo valor

de 920 kv corresponde a una constante que relaciona la cantidad de RPM por cada voltaje consumido.

FORMATO



DIAMETRO DE LA HELICE

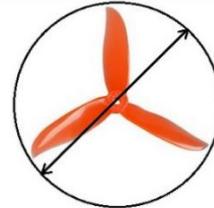


Ilustración 2.5 Nomenclatura y formato de hélices [15]

Las hélices son componentes importantes y su funcionamiento se basa en la propulsión por rotor, que utiliza el principio de la acción y la reacción de Newton (también conocido como la ley de acción y reacción). Cuando una hélice gira, produce un flujo de aire hacia abajo, lo que genera una fuerza de ascenso permitiendo que el dron se eleve o se mueva hacia arriba y hacia abajo.

La nomenclatura de las hélices mostradas en la ilustración 2.5 se representa por 4 dígitos, los primero 2 indican el diámetro, los otros dos siguientes indican el paso es decir la distancia en vertical que avanza una hélice en una rotación completa. Por ejemplo, una hélice con código 5045 indica que el diámetro de punta a punta es de 50 cm mientras que el 45 indica la distancia vertical en milímetros por cada rotación [15].

2.8 Selección de la batería

Según la Ilustración 2.6 en donde se muestra las características de una batería. Se debe asegurar que la batería escogida sea capaz de suministrar la corriente necesaria para los motores; se revisó en la tabla de empuje y usando la ecuación 2.13 se calcula cuál será la corriente máxima del motor elegido funcionando al 100% de su potencia.

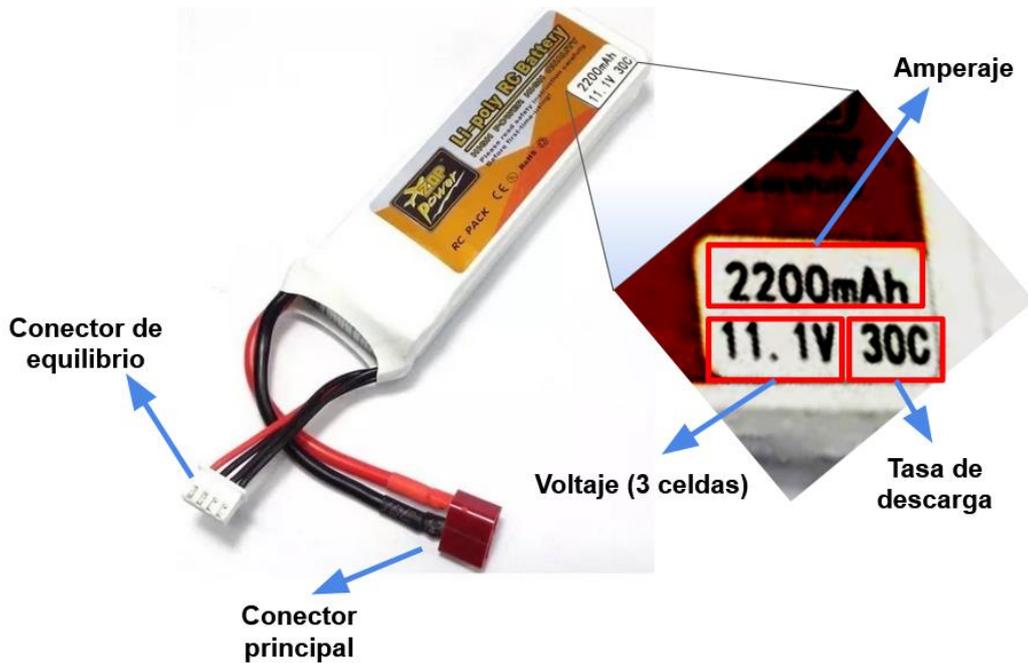


Ilustración 2.6 Características de una batería. Elaboración propia

Según la tabla 2.6, un motor 2212 920kv en conjunto con las hélices y batería seleccionadas anteriormente tiene una corriente máxima I_{motor} de 7.6 A usando unas hélices de 9.5 pulgadas con una batería de 11.1V.

$$I_{mmax} = I_{motor} \times n_{motor} \quad 2.13$$

$$I_{mmax} = 7.6 \times 4$$

$$I_{mmax} = 30.4 \text{ A}$$

Basándonos en la ecuación 2.13 La corriente máxima que consumirá los 4 motores es de 30.4 amperios, para saber si la batería elegida podrá suministrar esa cantidad de corriente es necesario asegurarse que su corriente máxima I_{bmax} sea mayor que la corriente máxima de los motores I_{mmax} : esta la podemos verificar mediante la ecuación 2.14 y comprobar con la ecuación 2.15

$$I_{bmax} = I_{hora} \times C_D \quad 2.14$$

$$I_{bmax} = 2.2 \times 30$$

$$I_{bmax} = 66 \text{ Ah}$$

$$I_{bmax} > I_{mmax} \quad 2.15$$

$$66 \text{ A} > 30.4 \text{ A}$$

Donde:

I_{bmax} : Corriente máxima de la batería.

I_{hora} : Corriente que suministra durante una hora [mAh].

C_D : Capacidad de descarga.

Luego, se considera la potencia total de la batería y el consumo de potencia del motor al 50% de su eficiencia usando las ecuaciones 2.16 y 2.17:

$$P_{bateria} = I \times V \quad 2.16$$

$$P_{bateria} = 2.2 \times 11.1$$

$$P_{bateria} = 24.42 \text{ W}$$

$$\text{Consumo}_{motor} = \frac{W_{dron}}{ef_{50\%}} \quad 2.17$$

$$\text{Consumo}_{motor} = \frac{580 \text{ gr}}{11.5 \text{ gr/W}}$$

$$\text{Consumo}_{motor} = 50.43 \text{ W}$$

Donde:

$P_{bateria}$: potencia del batería medido en watts

W_{dron} : peso total del dron medido en gramos.

$ef_{50\%}$: eficiencia de los motores al 50% de su potencia, este dato es obtenido de las especificaciones del motor o de las tablas de empuje.

Finalmente se calcula el tiempo de vuelo en horas usando la expresión 2.18

$$Tiempo_{vuelo} = \frac{P_{bateria}}{Consumo_{motor}} \quad 2.18$$

$$Tiempo_{vuelo} = \frac{24.42}{50.43} \times 60$$

$$Tiempo_{vuelo} = 29 \text{ min}$$

La ecuación 2.16 también puede ser representada de forma diferente para calcular el número de celdas que debe tener la batería en base a su duración de vuelo y los motores a utilizar, reemplazando la potencia de la batería y el consumo del motor por las expresiones 2.17 y 2.18 obtendremos la ecuación 2.19:

$$P_{bateria} = Tiempo_{vuelo} \times Consumo_{motor} \quad 2.19$$

$$I \times V = Tiempo_{vuelo} \times \frac{W_{dron}}{ef_{50\%}}$$

Mediante el despeje y reemplazo podemos obtener el voltaje de la batería como se puede apreciar en la ecuación 2.20:

$$V = \frac{Tiempo_{vuelo} \times W_{dron}}{I \times ef_{50\%}} \quad 2.20$$

El voltaje de la batería está ligado a su manufactura y de forma estandarizada, su voltaje total será igual al voltaje nominal por el número de celdas que contiene dando como resultado la ecuación 2.21:

$$V_n \times n_{celdas} = \frac{Tiempo_{vuelo} \times W_{dron}}{I \times ef_{50\%}}$$

$$n_{celdas} = \frac{\text{Tiempo}_{\text{vuelo}} \times W_{\text{dron}}}{V_n \times I \times ef_{50\%}}$$

Donde:

V_n : voltaje nominal de una celda de la batería.

n_{celdas} : número de celdas que posee una batería.

2.6 Diseño 3D en OpenScad

El diseño del fuselaje del dron es una tarea crucial para garantizar la estabilidad y seguridad del vehículo volador. Para lograrlo, se seguirá un proceso dividido en tres etapas: en la primera, se dibujará el cuerpo del dron, que servirá de soporte y protección para los componentes internos como la tarjeta controladora y las baterías. En la segunda etapa, se dibujarán los brazos del dron, que serán los encargados de sostener los motores y resistir las fuerzas generadas por el empuje de las aspas. Por último, se diseñará una base para fijar los rotores a los brazos, que se personalizará para adaptarse a diferentes medidas de rotores como se observa en la Ilustración 2.7. De esta forma, se logrará un diseño del fuselaje del dron que permita su parametrización y garantice un vuelo estable y seguro.

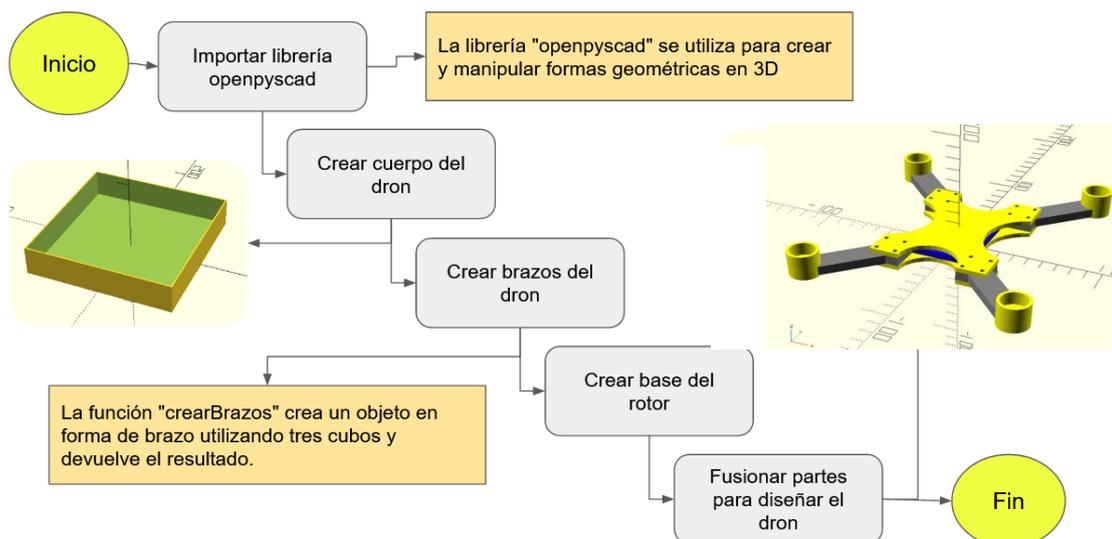


Ilustración 2.7 Metodología del diseño de dron

2.6.1 Diseño mecánico del cuerpo

El diseño del cuerpo es una tarea crucial para garantizar la estabilidad y seguridad del vehículo volador. El diseño debe tener en cuenta la forma y tamaño del dron, el peso y tamaño de los componentes internos como la batería y las fuerzas externas a las que estará expuesto durante el vuelo.

Para poder crear la pieza del cuerpo en OpenScad usando Python hay que considerar que este software solo ofrece la creación de objetos 3D de figuras básicas como: un cubo, cilindros y esferas.

El catálogo de figuras posibles es limitado, por lo que para realizar figuras más complejas se debe jugar con las operaciones booleanas de una o más de estas figuras simples, en la Ilustración 2.8 se muestra cómo se fue diseñado la parte superior del dron partiendo de una extrusión de una figura rectangular plana y luego se realizó una resta o una operación booleana de vaciado con 4 cilindros.

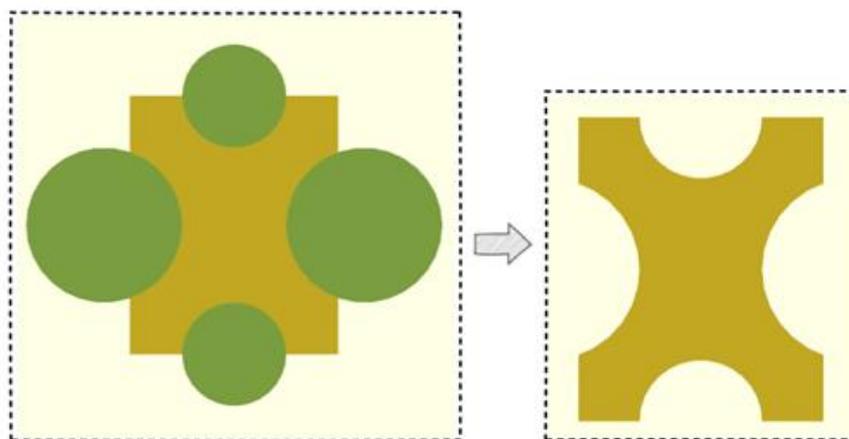


Ilustración 2.8 Pasos para el diseño del dron base

Es preciso asegurarse de que la pieza soportará los esfuerzos a los que estará sometida, por lo que para realizar su cálculo estructural hay que dimensionar correctamente los parámetros como el grosor, base y altura; si el grosor de su estructura es demasiado delgado este no soportará el peso de los componentes llegando a su punto de rotura.

Para dimensionar correctamente aproximaremos la sección de la figura que será la que mayor carga estará sometida, considerándola como una barra doblemente empotrada, así como se observa en la Ilustración 2.9.

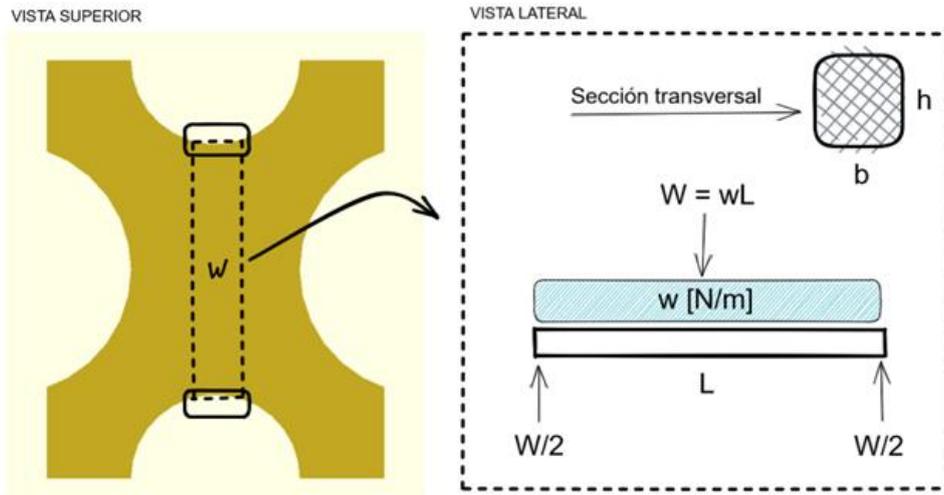


Ilustración 2.9 Sección de barra a analizar, Vista lateral del cuerpo aproximándola a una viga doblemente empotrada.

La barra mostrada en la Ilustración 2.10 soporta una carga uniformemente distribuida a lo largo de su longitud, el diagrama cortante como se observa en la figura 12 muestra que por simetría su tendrá su máximo cortante en los extremos y su momento máximo en el centro.

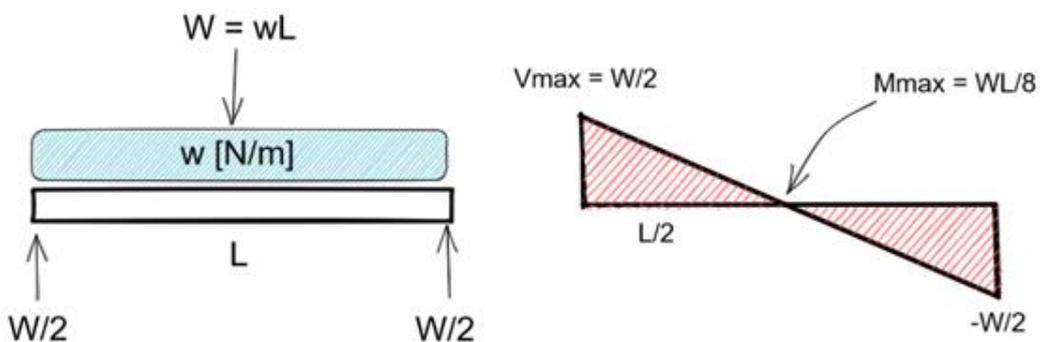


Ilustración 2.10. Diagrama de esfuerzo cortante

Se calcula la carga admisible con la expresión matemática 2.19:

$$\tau_{max} = \frac{3V}{2bh} \quad (2.19)$$

Si conocemos su esfuerzo máximo, la base y la altura de la sección transversal de la viga podremos calcular el peso máximo que podrá soportar.

2.6.2 Diseñar brazos del dron

Para diseñar los brazos que sostienen los rotores en un dron, es importante considerar una serie de factores, como la capacidad de carga, la resistencia a la flexión y la resistencia al viento. Además, es importante tener en cuenta el peso total del dron, incluyendo todos los componentes y cargas, para asegurarse de que los brazos sean lo suficientemente fuertes para sostener el dron en el aire.

En general, se recomienda utilizar materiales resistentes y ligeros para los brazos, como fibra de carbono o aluminio. También es importante utilizar un diseño que distribuya de manera equitativa la carga entre los brazos, para asegurarse de que el dron sea estable en el aire.

Para determinar las dimensiones y el grosor de los brazos de la Ilustración 2.11. Se puede utilizar un software de diseño o cálculos estructurales manuales para determinar la capacidad de carga y la resistencia a la flexión necesarias. También es importante realizar pruebas de vuelo para asegurarse de que el diseño funcione adecuadamente y permita al dron volar de manera estable.

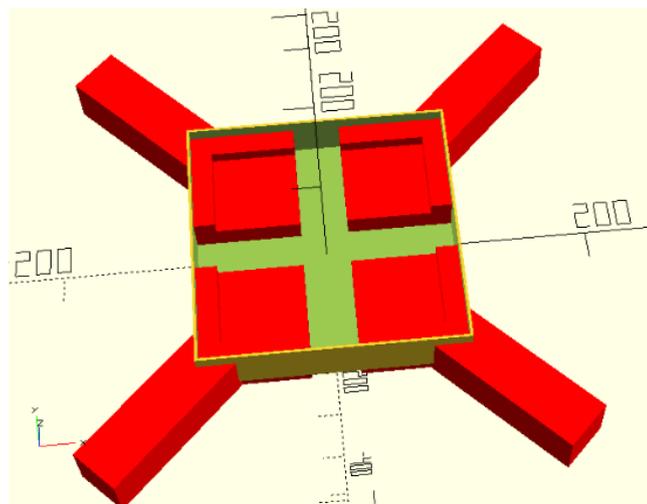


Ilustración 2.11 Modelo de dron generado con código. Elaboración propia

CAPÍTULO 3

3. RESULTADOS Y ANÁLISIS

En el presente capítulo se presentará un diseño 3D de un dron pequeño, el cual se construyó con el software de diseño de dron como se observa en la Ilustración 3.1, basado en el lenguaje de programación **Python**, y el software CAD **OpenScad**. Se mostrará el diseño del dron con todas sus piezas, el cual también se lo importará a un formato STL como se muestra en la Ilustración 3.2 y poder realizar un análisis de elementos finitos, para los brazos, la base y la base de los motores.

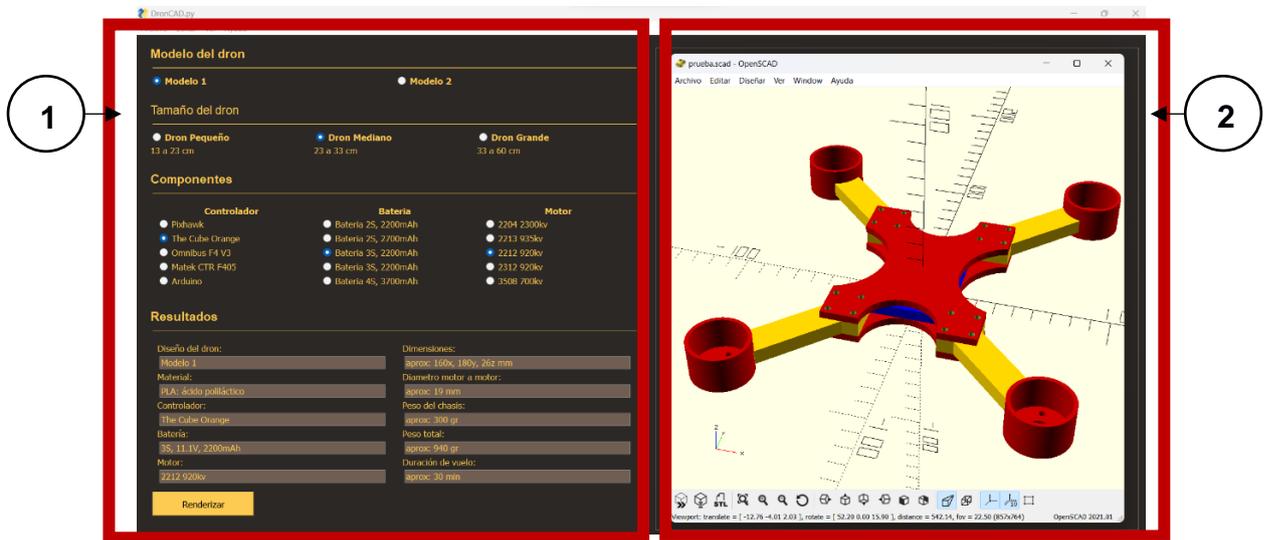


Ilustración 3.1 Esbozo de software encargado de diseñar un dron clase cuadróptero: (1) Sección la cual ingresarán las especificaciones del dron tales como dimensiones, aplicaciones, controlador entre otras. (2) sección donde se visualiza el modelo de manera Automático

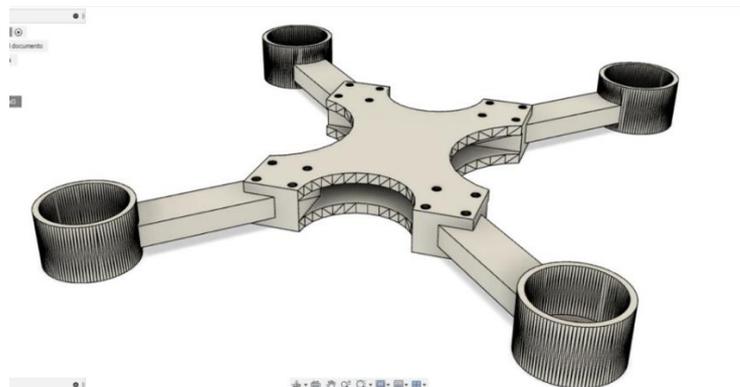


Ilustración 3.2. Diseño 3D del dron obtenido en formato STL

3.1 Diseño de Software

3.1.1 Front-end

En la Ilustración 3.3, se ilustra la interfaz del proyecto propuesto, mismo que fue desarrollado con el lenguaje de programación Python y el software CAD OpenScad. Ya que, este programa permitió modelar el dron, mediante las librerías de Python.

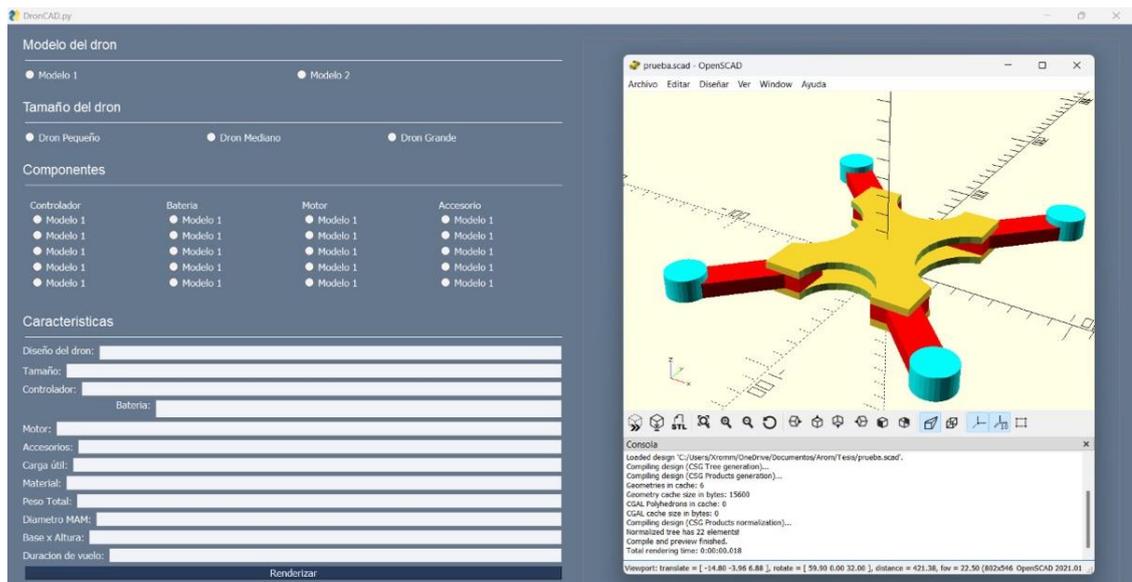


Ilustración 3.3 Interfaz gráfica del proyecto

3.1.2 Back-end

En esta sección se muestra el desarrollo del Back-end, en donde se importó las librerías **numpy** (usada para las ecuaciones matemáticas del diseño del dron), **pyOpenscad** (sirve para poder controlar el software OpenScad mediante Python), **simplepygui** (para crear la interfaz gráfica del software).

En la Ilustración 3.4 y la Ilustración 3.5 se realizaron funciones, las cuales fueron necesarias para la creación de componentes, como las dimensiones del dron, siendo esta la principal, la elección de la placa, la batería entre otros componentes electrónicos y/o mecánicos.

```

Edit Selection View Go Run Terminal Help
1 prueba.py 2 tabla.py 3 prueba2.py 4 prueba3.py X
C:\Users\Xromm > OneDrive > Documentos > Arom > Tesis > 4 prueba3.py > ...
1 import win32gui #Me permite controlar el comportamiento de las aplicaciones de windows
2 import win32con #Me permite controlar el comportamiento de las aplicaciones de windows
3 import win32api
4 import PySimpleGUIQt as sg #Me permite hacer aplicaciones con interfaz grafica
5 import threading #Me permite manejar hilos
6 import time #libreria con herramientas relacionadas al tiempo
7 import pyautogui #Me permite controlar los perifericos de entrada y salida
8 import multiprocessing #similar al de hilos pero con procesos
9 import subprocess
10 import openpyscad as ops
11
12 #####Diseño de componentes####
13 > def crearDimensiones(base, altura, grosor): #Esto solo ayuda a crear el cuerpo del dron---
20
21 > def crearBateria(base, altura, grosor):---
26
27 > def crearMotores():---
29
30 > def crearPlaca():---
32
33 > def crearCamara(): ---
35
36 > def crearGPS(): ---
38
39 > def crearCardan(): ---
41
42 #####Diseño del dron####
43 def crearDimensiones(base, altura, grosor): #Esto solo ayuda a crear el cuerpo del dron
44     cuboFrente = ops.Cube([base/2, base/2, grosor+5], center = True)
45     cuboFrente = cuboFrente.translate([0,altura/2,grosor/2+1])
46     cuboDer = ops.Cube([altura - base/2, altura - base/2, grosor+5], center = True)
47     cuboDer = cuboDer.translate([(altura - base/2)/2+base/2-base/4,0,grosor/2+1])
48     dimensiones = cuboFrente + cuboDer
49     return dimensiones
50

```

Ilustración 3.4. código en Python, importaciones de librerías, definiciones de funciones

```

51
52
53 def crearCuerpoSuperior(base, altura, grosor):
54     cubo = ops.Cube([base, altura, grosor], center = True)
55     cubo = cubo.translate([0,0,grosor/2])
56
57     circuloFrente = ops.Cylinder(r=base/4, h=grosor+5)
58     circuloFrente = circuloFrente.translate([0,altura/2,-1])
59     circuloAtras = circuloFrente.mirror([0,1,0])
60
61     circuloDer = ops.Cylinder(r=(altura - base/2)/2, h=grosor+5)
62     circuloDer = circuloDer.translate([(altura - base/2)/2+base/2-base/4,0,-1])
63     circuloIzq = circuloDer.mirror([1,0,0])
64
65     cuerpo = cubo-circuloFrente-circuloAtras-circuloDer-circuloIzq
66     cuerpo = cuerpo.translate([0,0,grosor + base*0.1])
67
68     return cuerpo
69
70 def crearCuerpoInferior(base, altura, grosor):
71     cubo = ops.Cube([base, altura, grosor], center = True)
72     cubo = cubo.translate([0,0,grosor/2])
73
74     circuloFrente = ops.Cylinder(r=base/4, h=grosor+5)
75     circuloFrente = circuloFrente.translate([0,altura/2,-1])
76     circuloAtras = circuloFrente.mirror([0,1,0])
77
78     circuloDer = ops.Cylinder(r=(altura - base/2)/2, h=grosor+5)
79     circuloDer = circuloDer.translate([(altura - base/2)/2+base/2-base/4,0,-1])
80     circuloIzq = circuloDer.mirror([1,0,0])
81
82     cuerpo = cubo-circuloFrente-circuloAtras-circuloDer-circuloIzq
83
84     return cuerpo
85
86 def crearBrazo(base, altura, grosor):
87     dimensionx = base*0.2

```

Ilustración 3.5. funciones las cuales se encargan de dibujar en OpenScad

3.2 Diseño Mecánico

3.2.1 Brazos de dron

Según Ilustración 3.6, Ilustración 3.7 e Ilustración 3.8 se efectuó un análisis de fuerzas en los brazos del dron, mediante el método FEA. El objetivo fue simular a los brazos del dron como si fueran vigas empotradas, a las cuales se le someterá

a una fuerza, en la esquina (lugar donde ira el motor), y así observar el comportamiento de los brazos bajo una carga, con el fin de identificar posibles problemas de diseño.

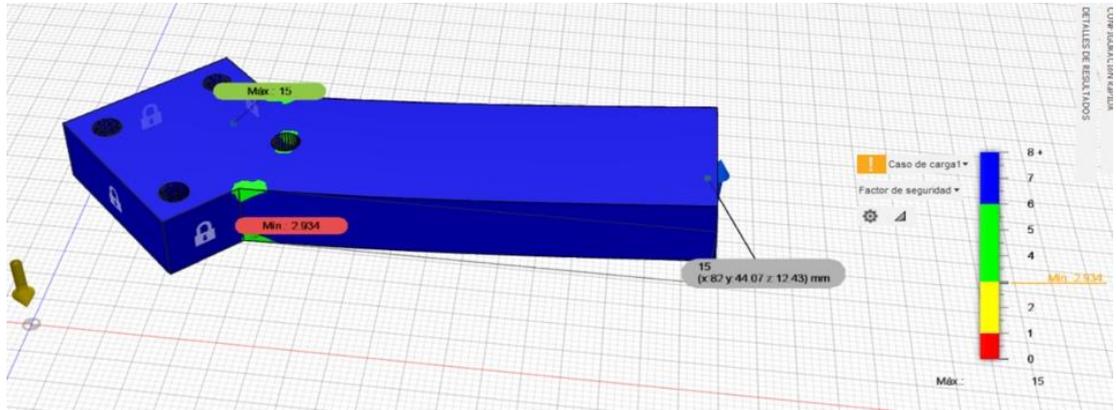


Ilustración 3.6. Factor de seguridad

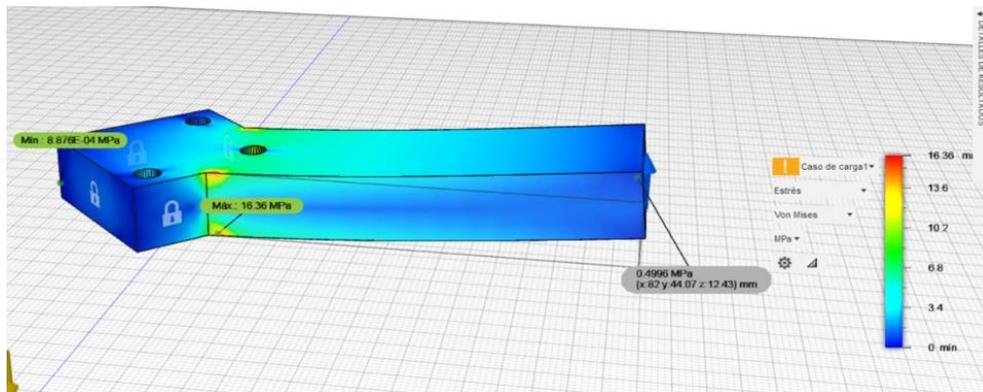


Ilustración 3.7. Análisis de fatiga por Von mises

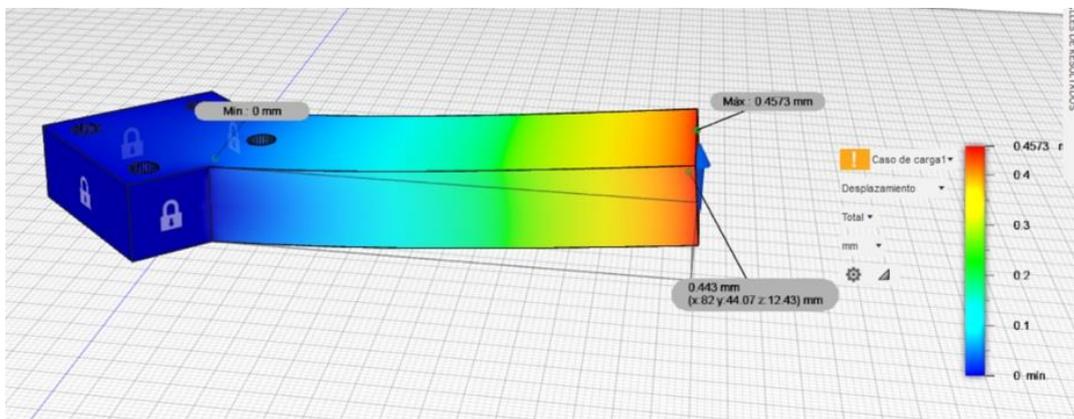


Ilustración 3.8. Análisis de tensión estática

Los resultados de dicho análisis indicaron que en el brazo se alcanza unos esfuerzos de hasta 0.5 Mpa aproximadamente, de igual forma podemos observar que se vendría deformar unos 0.43 mm por el esfuerzo ascendente al cual están sometidos por el empuje del motor, sin embargo, observamos que tenemos un buen factor de seguridad en el diseño del dron.

3.2.2 Base de motores

Se realizó un análisis de fuerzas en las bases de los motores, mediante el método FEA. En la base antes mencionada se la sometió a una fuerza en el centro (donde se colocará el motor), y así observar el comportamiento de la base, con el fin de identificar posibles problemas del diseño obtenido por el software.

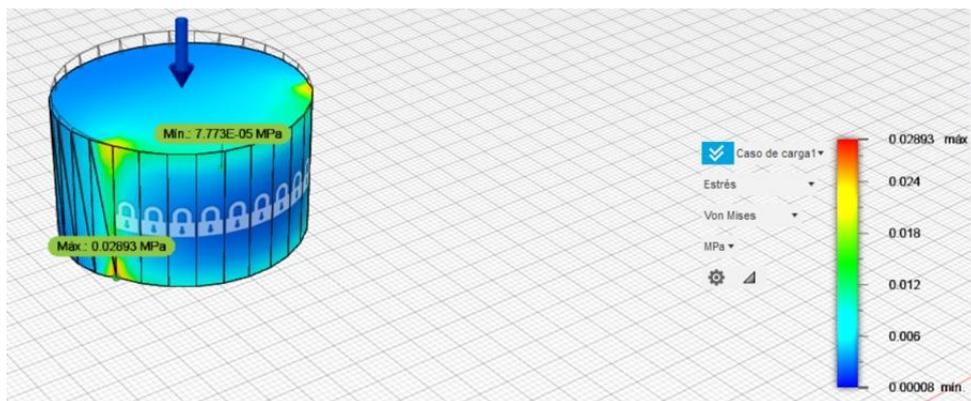


Ilustración 3.9 de fatiga por Von mises, se toma en cuenta solo el peso del motor.

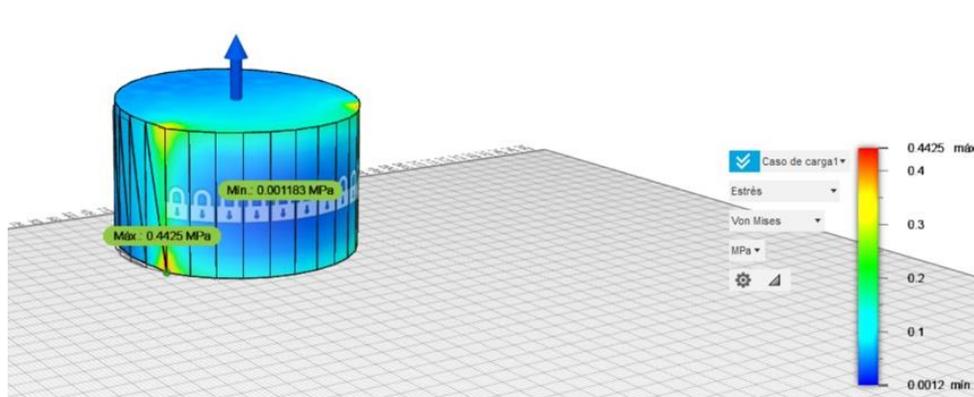


Ilustración 3.10. Análisis de fuerza, considerando la fuerza generada por el motor.

Los resultados del análisis indicaron que la base construida con un material de PLA, la cual se sometió a un peso de 0,5 N por el peso del motor, y posterior

cuando estos están a máxima potencia considerando la fuerza que estos generan que es equivalente a 10N, obteniendo esfuerzos máximos de 0.0234 Mpa

3.2.3 Dron

Se desarrolló un análisis de fuerzas en el dron, mediante el método FEA. En el diseño del dron obtenido mediante el software, con el fin de identificar posibles problemas del diseño, y poder rediseñar o mejorar el dron. Como se observa en la Ilustración 3.12 en donde se obtiene el esfuerzo de von mises y analizar el comportamiento del material antes este esfuerzo. De igual forma en Ilustración 3.13



Ilustración 3.11 Factor de seguridad

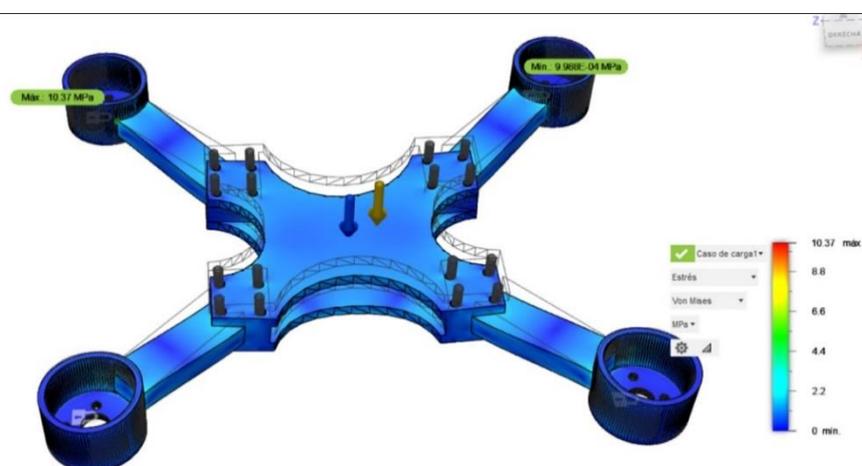


Ilustración 3.12 Análisis de fatiga por Von Mises.

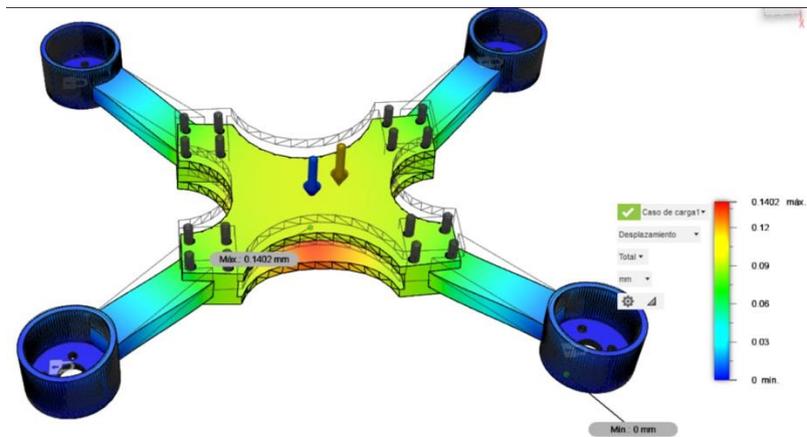


Ilustración 3.13 Análisis de tensión estática.

Los resultados del análisis indicaron que el dron construido con PLA (ácido poli láctico), tuvo unos esfuerzos máximos 0.4475 MPa, de igual forma, una vez puestas los componentes electrónicos, obtuvimos que este se puede llegar a deformar más en el centro debido al peso el cual fue sometido.

3.3 Análisis de Costos

Según la Tabla 3.1 para el análisis de costo del software se consideró, el costo de desarrollo de software, el cual incluyó los costos de contrato de un desarrollador de software junior, y un diseñador de drones; costos de pruebas el cual se encargó de probar el software en diferente entorno validando el rendimiento, marketing y publicidad, finalmente el costo de producción del dron el cual consistió en la impresión en 3D del dron.

Decripcion	cantidad	Precio Unitario	unidad	Precio Total	
Salario de Ing. de Software Jr	3		mes	\$1.100,00	\$3.300,00
Licencias de Programa	1		u	\$0,00	\$0,00
Mano de Obra(ensamblaje de dron)	1		u	\$20,00	\$20,00
Soporte tecnico y asesoria	1		u	\$150,00	\$150,00
Pruebas de rendimiento	1		u	\$100,00	\$100,00
					\$3.570,00

Tabla 3.1. Análisis de costos

.....

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- El software desarrollado cumple con el objetivo de renderizar, diseñar y modelar un dron tipo cuadricóptero, mediante los parámetros ingresados por el usuario. A través del uso del lenguaje de programación Python en conjunto con el software Openscad, el cual se tiene acceso a este por medio de las librerías de Python.
- Para el dimensionamiento del dron, se usaron las fórmulas y relaciones correspondientes, como se puede observar en la sección 2.6.1 diseño mecánico del chasis el cual, Generará un chasis mediante los datos ingresados de la altura, grosor, y ancho.
- La validación del prototipo virtual se realiza de manera efectiva, ya que se garantiza que todas las partes del dron se encuentren correctamente ensambladas y visualizadas en otros softwares de diseño 3D. Esto permite que el usuario pueda tener una vista previa de su diseño antes de la fabricación real, optimizando el tiempo y los costos en la producción del dron.
- El costo final del software de diseño de drones es de \$3570 dólares americanos, a pesar de esto el valor se encontraría por debajo de la manufactura del diseño de un dron esto es debido que para crear el diseño de un dron se debe pagar a un equipo de trabajo en diseño mecánico, electrónico, junto con la investigación que hay detrás para el desarrollo de estos productos.

Recomendaciones

- Se recomienda realizar más diseños en 3D para que haya mayor variedad al momento de construir un fuselaje incluyendo drones con múltiples rotores y drones de ala fija, así los diseños abarcarían mayores campos de aplicación lo cual sería perfecto para que los usuarios puedan elegir el modelo que se adapte de mejor manera a sus necesidades.
- El software es capaz de exportar los diseños en formatos compatibles con la impresión 3D, por lo que se recomienda programar una función que sea capaz de recopilar todas las piezas y realizar un solo bloque de impresión optimizando el espacio y el tiempo de la impresora, de esta manera se puede imprimir todas las piezas que conforman el dron con una sola impresión.
- Dentro del código se encuentran las funciones, cálculos y ecuaciones necesarias para que el proyecto pueda ser escalado a niveles inimaginables, por lo que se recomienda realizar actualizaciones para dotar de nuevas funcionalidades y nuevas características únicas que lo diferencien de otros software como la recopilación de datos de entrada y los de salida para posterior a eso, realizar un modelo de entrenamiento para una inteligencia artificial, que pueda generar diseños 3D de forma más eficiente.

BIBLIOGRAFÍA

- [1] D. G., "Introductory Chapter: Drones," *Drones - Applications*, doi: 10.5772/intechopen.76943., 2018.
- [2] A. p. d. drones, «IDC App de drones,» 2020. [En línea]. Available: <https://idc.apddrones.com/drones/drones-multirotor-vs-ala-fija/>.
- [3] K. M. Z. S. Z. W. C. Bonilla Yoza Maria Mercedes, «EL USO DE LOS DRONES EN EL ÁMBITO PROFESIONAL,» *Publicación cuatrimestral UNESUM-Ciencias*, pp. 75-82, 2021.
- [4] G. d. Ecuador, «Ecuador ya cuenta con un reglamento para el uso de drones,» 24 noviembre 2020.
- [5] E. UNIVERSO, «El primer centro para operar drones se abre en Quito,» *EL UNIVERSO*, pp. 1-2, 14 mayo 2021.
- [6] R. d. s. 360, «¿Cuáles son las diferentes partes de un dron?,» *Revista de seguridad 360*, 16 Noviembre 2022.
- [7] DJI, «DJI,» 2019. [En línea]. Available: https://www.dji.com/mini-2?site=brandsite&from=mobile_landing_page.
- [8] DronProfesional, «DronProfesional.com Video Areo,» DronProfesional, 23 Marzo 2018. [En línea]. Available: <https://dronprofesional.com/blog/cuales-son-las-partes-de-un-dron/>. [Último acceso: 2022 Noviembre 16].
- [9] «Amazon,» 2022. [En línea]. Available: <https://www.amazon.com/-/es/2212-920-KV-Escobillas-Quadcopter-Hexacopter-Multicopter/dp/B01LWXA717>.
- [10] M. L. Mendoza, «OpenWebinars,» Julio 2020. [En línea]. Available: <https://openwebinars.net/blog/que-es-un-lenguaje-de-programacion/>.
- [11] PLATZI, «platzi,» 2020. [En línea]. Available: https://platzi.com/blog/python/?utm_source=google&utm_medium=cpc&utm_campaign=19643931773&utm_adgroup=&utm_content=&gclid=CjwKCAiA0JKfBhBIEiwAPhZXD6imOc6iRZBRoiQR6bYXxAJ1XnZHFz4q7IAwx5oHULp9fWjp3L-iEhoCHdIQAvD_BwE&gclid=aw.ds.

- [12] M. G. Talavera, «CIENCIA UANL,» 26 Enero 2016. [En línea]. Available: <https://cienciauanl.uanl.mx/?p=5321>.
- [13] A. P, «Directory of U.S. Military Rockets and Missiles,» 2006. [En línea]. Available: <http://www.designation-systems.net/dusrm/app2/q-1.html>.
- [14] L. P, «NEOTEO,» 2019. [En línea]. Available: <https://www.neoteo.com/como-se-fabrican-los-drones/>.
- [15] C. FPV, «Chicken FPV,» 2019. [En línea]. Available: <https://chickenfpv.com/chickenfpv/los-mejores-componentes-para-drones-de-carreras-y-freestyle/las-mejores-helices-de-drones-de-carreras-y-freestyle/>.
- [16] CropLife, «Drones en agricultura, una tecnología con potencial para apoyar especialmente a los pequeños agricultores,» CropLife, 4 Noviembre 2021. [En línea]. Available: <https://www.croplifela.org/es/actualidad/articulos/drones-en-agricultura-una-tecnologia-con-potencial-para-apoyar-especialmente-a-los-pequenos-agricultores>. [Último acceso: 6 noviembre 2022].
- [17] E. DIARIO, «Los drones ahorran tiempo y dinero al agricultor en Ecuador y Colombia,» *EL DIARIO*, pp. 1-3, 24 Agosto 2022.
- [18] G. D, «Introductory chapter: Drones,» de *Application of drones*, 2022.

APÉNDICES

APÉNDICE A

Evaluación de alternativas para cada criterio}

Preferencia del Cliente	Alternativa 1 > Alternativa 2 > Alternativa 3				
Alternativa	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1	1	0,5	0,5	2	0,4
Alternativa 2	0,5	1	0	1,5	0,3
Alternativa 3	0	0	1	1	0,2
			suma	4,5	1,00

Ilustración A.1 Evaluación de alternativas ante el criterio “preferencia del cliente”

Costo	Alternativa 1 > Alternativa 2 = Alternativa 3				
Alternativa	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1	1	0,5	0,5	2	0,4
Alternativa 2	0	1	0,5	1,5	0,3
Alternativa 3	0	0,5	1	1,5	0,3
			suma	5	1,00

Ilustración A.2 Evaluación de alternativas ante el criterio “Costo”

Facilidad de Uso	Alternativa 2 > Alternativa 1 > Alternativa 3				
Alternativa	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1	1	0,5	0,5	2	0,333
Alternativa 2	0,5	1	1	2,5	0,417
Alternativa 3	0,5	0	1	1,5	0,250
			suma	6	1,00

Ilustración A.3 Evaluación de alternativas ante el criterio “Facilidad de uso”

Fiabilidad	Alternativa 2 > Alternativa 1 = Alternativa 3				
Alternativa	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1	1	0	0,5	1,5	0,3
Alternativa 2	0,5	1	0,5	2	0,4
Alternativa 3	0,5	0	1	1,5	0,3
			suma	5	1,0

Ilustración A.4 Evaluación de alternativas ante el criterio “Fiabilidad”

Flexibilidad	Alternativa 1 > Alternativa 2 = Alternativa 3				
Alternativa	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1	1	1	1	3	0,4
Alternativa 2	0,5	1	0,5	2	0,3
Alternativa 3	0,5	0,5	1	2	0,3
			suma	7	1,0

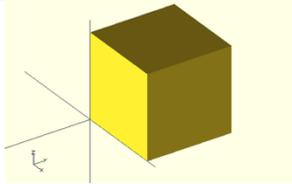
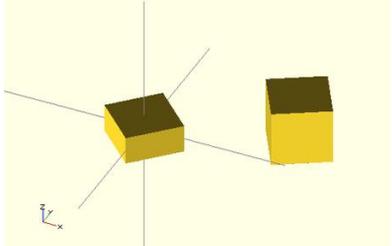
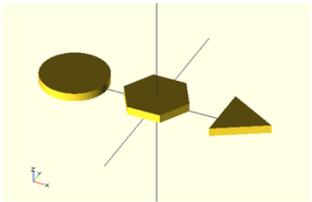
Ilustración A.5 Evaluación de alternativas ante el criterio “Flexibilidad”

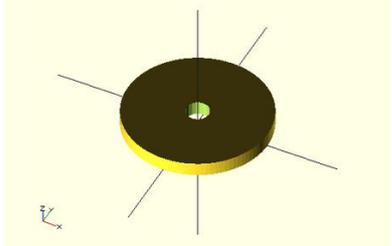
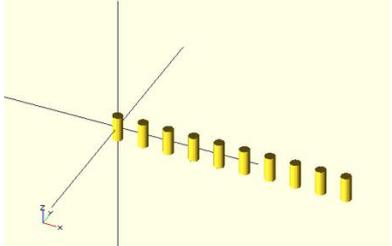
Escalabilidad	Alternativa 1 = Alternativa 3 > Alternativa 2				
Alternativa	Alternativa 1	Alternativa 2	Alternativa 3	$\Sigma+1$	Ponderación
Alternativa 1	1	0,5	0,5	2	0,4
Alternativa 2	0	1	0	1	0,2
Alternativa 3	0,5	0,5	1	2	0,4
			suma	5	1,0

Ilustración A.6 Evaluación de alternativas ante el criterio “Escalabilidad”

APÉNDICE B

Introducción al código de programación del software cad Openscad

Descripción	Programa	Figura
Cubo "Hola mundo"	<pre>//-- Cubo Hola Mundo cube([10,10,10]);</pre>	
Traslación y rotación	<pre>//-- Ejemplo de translación y rotación //-- Traslación y rotación de un cubo rotate([0,0,30]) translate([50,0,0]) cube([20,20,20], center=true); //-- rotación de un cubo rotate([0,0,45]) cube([20,20,10], center=true);</pre>	
Cilindros	<pre>//-- Moneda translate([-50,0,0]) cylinder(r=40/2, h=5, \$fn=100); //-- Hexágono cylinder(r=40/2, h=5, \$fn=6); //-- Triángulo equilátero translate([50,0,0]) cylinder(r=40/2, h=5, \$fn=3);</pre>	

Agujeros	<pre>//-- Rueda simple difference() { //-- Base de la rueda cylinder(r=50/2, h=5, \$fn=100); //-- Taladro de 8mm cylinder(r=8/2, h=20, \$fn=20, center=true); }</pre>	
Repetir tareas	<pre>drill=4; h1=10; d=10; n = 20; for (i = [0: n-1]) { translate([i*d,0,0]) cylinder(r=drill/2, h=h1, \$fn=20, center=true); }</pre>	

Código del software para generar fuselajes de drones

El código completo del software se encuentra en el repositorio <https://github.com/arommore/DronCAD> sin embargo, aquí se detalla secciones importantes para entender su arquitectura.

La programación hace uso de librerías propias que viene en el paquete estándar de instalación de Python, sin embargo, también se hace uso de librerías de terceros como PysimpleGUIQT [<https://github.com/PySimpleGUI/PySimpleGUI>] que da un conjunto de herramientas para facilitar la construcción de interfaz de usuario de una forma fácil, además incorpora la librería openpyscad [<https://github.com/taxpon/openpyscad>] usada para la construcción y renderización de figuras 3D.

```

1 import win32gui #Me permite controlar el comportamiento de las aplicaciones de windows
2 import win32con #Me permite controlar el comportamiento de las aplicaciones de windows
3 import PySimpleGUIQt as sg #Me permite hacer aplicaciones con interfaz grafica
4 import threading #Me permite manejar hilos
5 import time #Libreria con herramientas relacionadas al tiempo
6 import subprocess
7 import openpyscad as ops
8

```

En el apartado del diseño de componentes se crea las funciones para generar las figuras basándose en ciertos parámetros que las definen.

```

9 #####Diseño de componentes#####
10 def crearDimensiones(base, altura, grosor): #Esto solo ayuda a crear el cuerpo del dron
11     cuboFrente = ops.Cube([base/2, base/2, grosor+5], center = True)
12     cuboFrente = cuboFrente.translate([0,altura/2,grosor/2+1])
13     cuboDer = ops.Cube([altura - base/2, altura - base/2, grosor+5], center = True)
14     cuboDer = cuboDer.translate([(altura - base/2)/2+base/2-base/4,0,grosor/2+1])
15     dimensiones = cuboFrente + cuboDer
16     return dimensiones
17
18 def crearBateria(base, altura, grosor):
19     bateria = ops.Cube([base/2, altura - base/2, base/10], center=True)
20     bateria = bateria .translate([0,0,base/20 + grosor/2])
21     bateria = bateria.color("Blue")
22     return bateria
23
24 def crearPerno(diametro,altura):
25     perno = ops.Cylinder(r = diametro/2, h = altura, _fn=30)
26     return perno

```

En diseño de interfaz se genera toda la interfaz visual del código, así como bloques contenedores para una correcta distribución de la interfaz.

```

155 #####Diseño de interfaz#####
156 def mostrarInterfaz():
157
158     sg.theme("DarkAmber")
159
160     menu_def = [
161         ['Archivo', ['Abrir', "Cambiar Tema", 'Guardar', 'Guardar como...']],
162         ['Editar', ['Copiar', 'Pegar', 'Cortar']],
163         ['Ver', ['Ver barra de herramientas']],
164         ['Ayuda', ['Acerca de...']]
165     ]
166
167     #Columnas de Componentes
168     layoutControlador = [ #Columna de seleccion del controlador
169         [sg.Text('Controlador', font=("Helvetica", 10, "bold"), justification='center')],
170         [sg.Radio('Pixhawk', 3, key='controlador_1', change_submits=True)],
171         [sg.Radio('The Cube Orange', 3, key='controlador_2', change_submits=True)],
172         [sg.Radio('Omnibus F4 V3', 3, key='controlador_3', change_submits=True)],
173         [sg.Radio('Matek CTR F405', 3, key='controlador_4', change_submits=True)],
174         [sg.Radio('Arduino', 3, key='controlador_5', change_submits=True)],
175         #[sg.Text("Dimensiones: ", tooltip="Poner valores x,y,z"),sg.InputText(key='controladores')]
176     ]

```

Durante la construcción del dron, se hace uso de todas las funciones generadoras de componentes creadas en el apartado de diseño de componentes, para ir construyendo el dron en base a los resultados obtenidos de las ecuaciones y del dimensionamiento de los componentes elegidos.

```
339 #Construcción del dron|
340
341 base = 80
342 altura = 100
343 grosor = 4
344 diametroMotor = 28
345
346 ParteInferior = crearCuerpoInferior(base,altura,grosor)
347 ParteSuperior = crearCuerpoSuperior(base,altura,grosor)
348 Brazo1 = crearBrazo(base,altura,grosor)
349 Brazo2 = Brazo1.mirror([1,0,0])
350 Piernas = (Brazo1 + Brazo2).mirror([0,1,0])
351
352 baseMotor1 = crearBaseMotor(grosorPared=2,altura=20, diametro=diámetroMotor)
353 baseMotor1 = baseMotor1.translate([0,altura*0.5,0])
354 baseMotor1 = baseMotor1.rotate([0,0,-45])
355 baseMotor1 = baseMotor1.translate([base/2-base*0.2/2+diámetroMotor/2-4,altura/2-base*0.2/2+diámetroMotor/2-4,0])
356
357 baseMotor2 = crearBaseMotor(grosorPared=2,altura=20, diametro=diámetroMotor)
358 baseMotor2 = baseMotor2.translate([0,altura*0.5,0])
359 baseMotor2 = baseMotor2.rotate([0,0,45])
360 baseMotor2 = baseMotor2.translate([- (base/2-base*0.2/2+diámetroMotor/2-4),altura/2-base*0.2/2+diámetroMotor/2-4,0])
```

Existe un apartado de herramientas extras, aquí solo hay dos funciones principales, la una es para abrir la aplicación de OpenScad y mostrar el archivo del dron previamente generado y guardado por el código de Python, la segunda función permite expandir cualquier ventana presente en el escritorio de Windows.

```

391 #####Herramientas#####
392
393 def mostrarFigura():
394     # Especificamos el nombre de la aplicación y cualquier argumento necesario
395     app_name = "C:\Program Files\OpenSCAD\openscad.exe"
396     app_args = ["prueba.scad"]
397
398     # Ejecutamos la aplicación
399     subprocess.run([app_name] + app_args)
400
401 def expandirAplicacion(tiempoEspera, nomAplicacion):
402     #Espera un tiempo a que abra la aplicacion
403     time.sleep(tiempoEspera)
404     # Obtener la ventana de la aplicación que quieres maximizar
405     hwnd = win32gui.FindWindow(None, nomAplicacion)
406     print(hwnd)
407     # Maximizar la ventana
408     win32gui.ShowWindow(hwnd, win32con.SW_MAXIMIZE)
409     print("se debería agrandar")
410

```

Finalmente, el código inicia creando los hilos que se ejecutan de forma independiente y en paralelo, donde llaman a la función de mostrar la interfaz gráfica y el segundo hilo espera a que se muestra la interfaz para expandirla y hacer más cómoda su visualización.

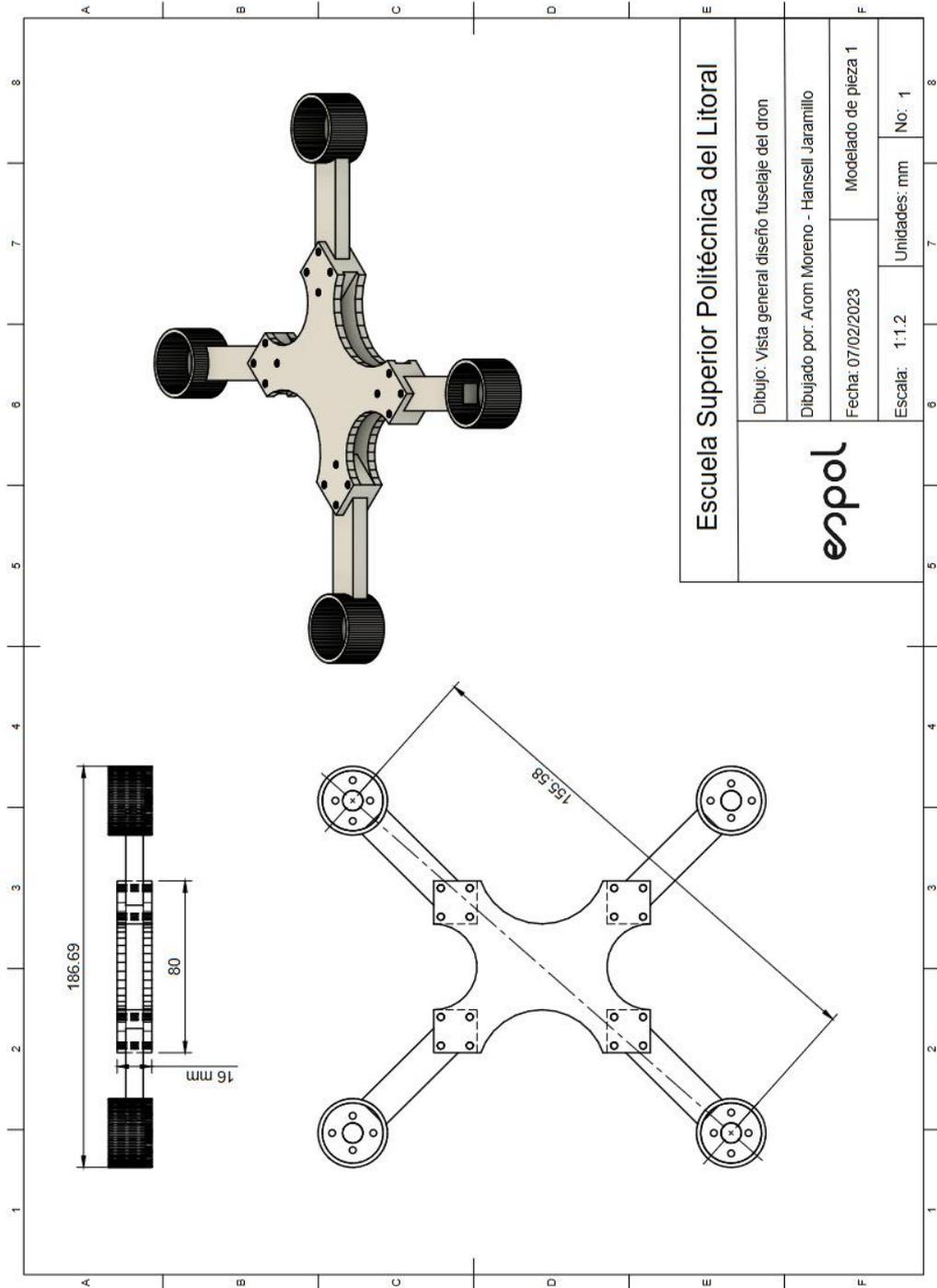
```

428 # Creamos una instancia de la clase Thread y le pasamos la función que queremos ejecutar
429 hilo1 = threading.Thread(target=mostrarInterfaz)
430 hilo4 = threading.Thread(target=expandirAplicacion, args=(2,"DronCAD.py"))
431
432 # Iniciamos el hilo
433 hilo1.start()
434 hilo4.start()

```

APÉNDICE C

Planos mecánicos



Escuela Superior Politécnica del Litoral

Dibujo: Vista general diseño fuselaje del dron

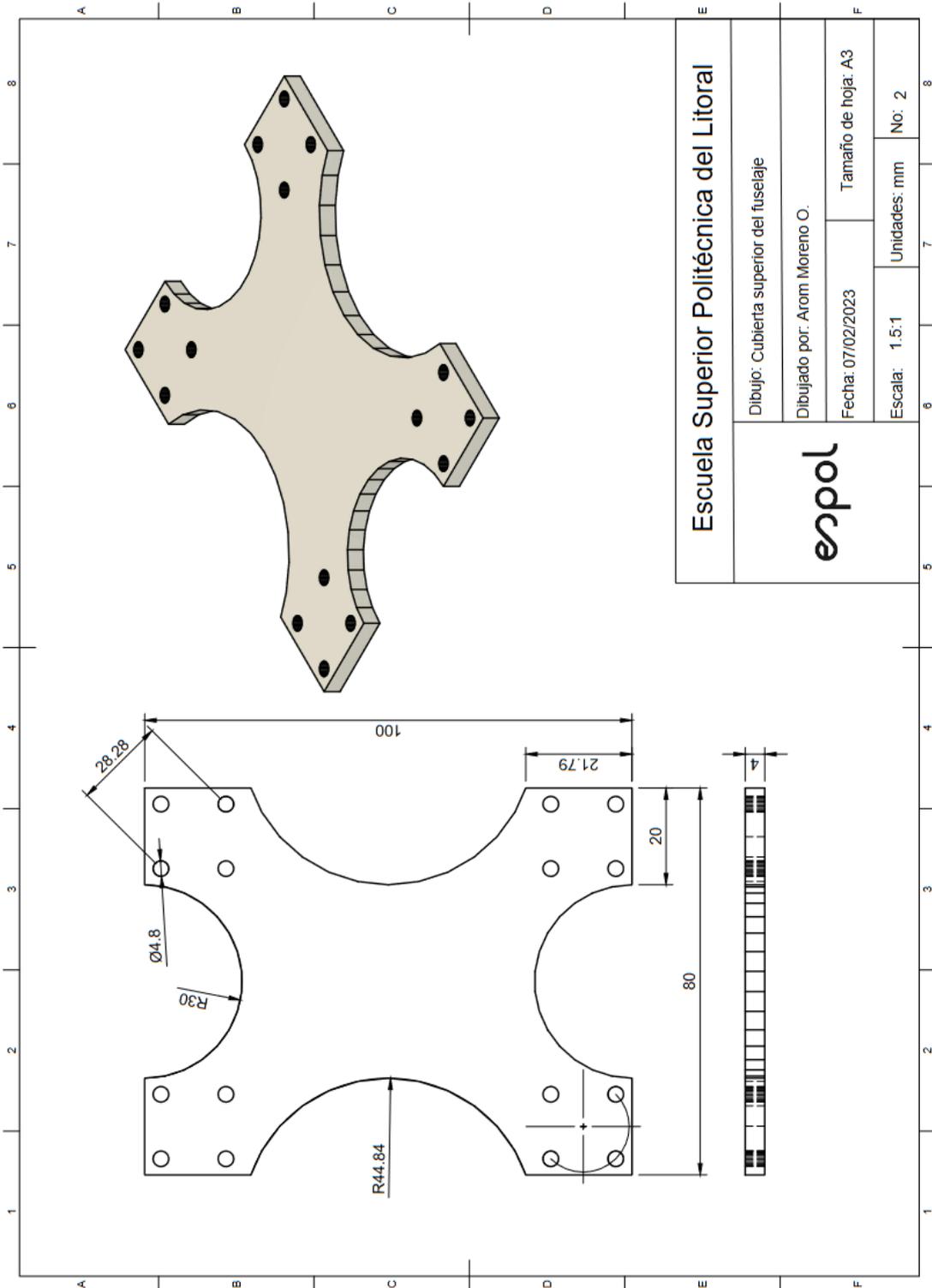
Dibujado por: Arom Moreno - Hansell Jaramillo

Fecha: 07/02/2023 Modelado de pieza 1

Escala: 1:1.2 Unidades: mm No: 1

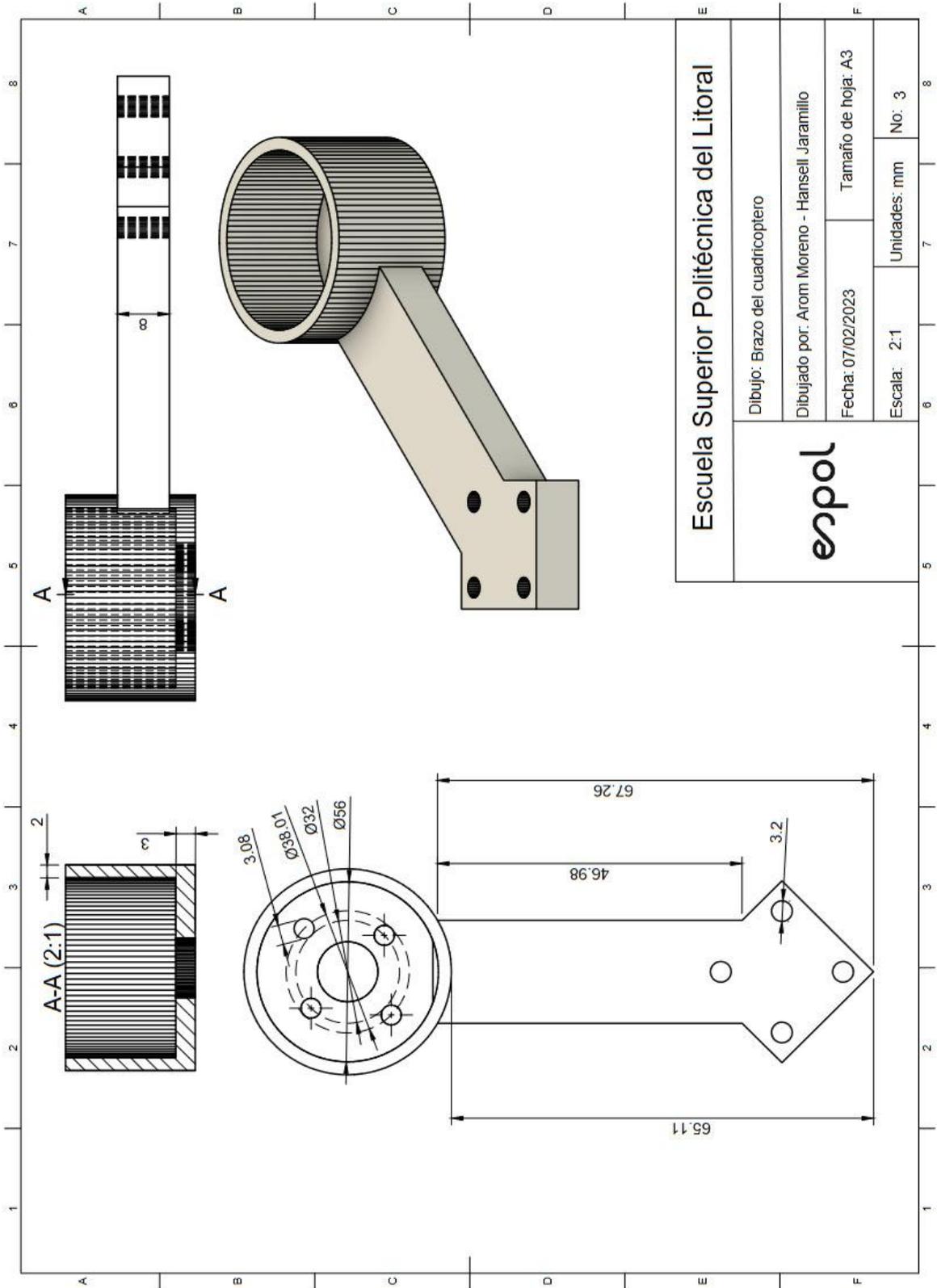


PLANO 1 Vista general diseño de fuselaje con dron



espol		Escuela Superior Politécnica del Litoral	
		Dibujo: Cubierta superior del fuselaje	
Dibujado por: Arom Moreno O.		Fecha: 07/02/2023	
Escala: 1.5:1		Unidades: mm	
No: 2		Tamaño de hoja: A3	

PLANO 2 Cubierta superior del fuselaje



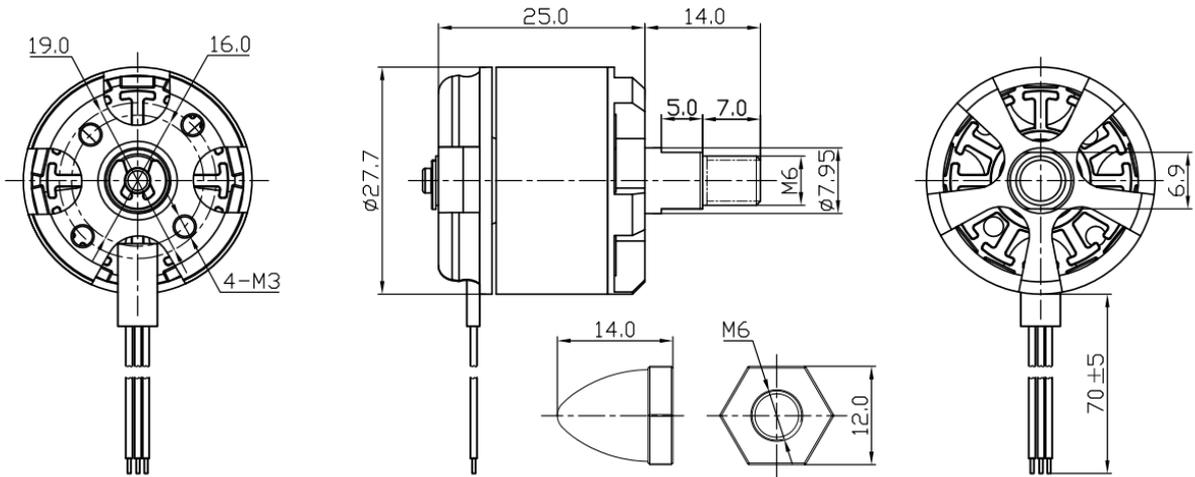
espol		Escala: 2:1		Unidades: mm	No: 3
		Fecha: 07/02/2023		Tamaño de hoja: A3	
Dibujo: Brazo del cuadricoptero		Dibujado por: Arom Moreno - Hansell Jaramillo			
Escuela Superior Politécnica del Litoral					

PLANO 3 Brazo del cuadricóptero

APÉNDICE D

Especificaciones sobre el motor seleccionado

MOTOR OUTLINE DRAWING



MOTOR PERFORMANCE DATA

MODEL	KV (rpm/V)	Voltage (V)	Prop	Load Current (A)	Pull (g)	Power (W)	Efficiency (g/W)	Lipo Cell	Weight (g) Approx
B2212	920	11.1	8045	7.3	465	81	5.7	2-4S	50
			1045	9.5	642	105	6.1		
	980	11.1	8045	8.1	535	90	5.9		
			1045	10.6	710	118	6.0		