

Escuela Superior Politécnica del Litoral

Facultad de Ingeniería en Mecánica y Ciencias de la Producción

Diseño e integración de un simulador de navegación para la formación y
práctica del personal militar en flotas marinas

INGE-2309

Proyecto Integrador

Previo la obtención del Título de:

Ingenieros en Mecatrónica

Presentado por:

André Alberto Aguirre Apolo

Alexandra Elizabeth Imaicela Cañizares

Guayaquil – Ecuador

Año: 2023

Dedicatoria

Dedico este logro a los dos pilares en mi vida:
a mi amada madre, sacrificada e
inquebrantable, cuyo apoyo en todos los
ámbitos me brindó la confianza y fortaleza
para seguir adelante en este proceso de
aprendizaje y formación. Así mismo, su
ejemplo de lucha frente a una de las
enfermedades más graves me mostró que no
hay imposibles.

A mi querido padre, cuyos consejos fueron
para mí un gran apoyo cuándo más lo
necesité. Su entrega y sacrificio por darme
todo lo posible para llegar hasta donde estoy
ahora me enseñó que las cosas se construyen
con esfuerzo y dedicación.

André Aguirre

Dedicatoria

Este trabajo se lo dedico a Dios, por todas las bendiciones que me ha dado a lo largo de mi vida.

A mi mamá y a mis papás, por todo el tiempo dedicado para mi educación. A mi hermana Belén y mi tía Ana, que me han aconsejado cuando me he encontrado desmotivada.

A mi enamorado, por su apoyo incondicional en los momentos más angustiantes de mi carrera universitaria.

A mi mejor amiga, Angélica, ha sido como una hermana y a más de ello, he podido contar con ella en todo momento.

Finalmente, al M.Sc Efraín Terán y al Ph.D Marcelo Fajardo, sus recomendaciones y supervisión me han dado el conocimiento adecuado para cumplir las metas de este proyecto.

Alexandra Imaicela

Agradecimientos

Gracias A Dios, por la sabiduría y fortaleza a lo largo de la carrera. A mi familia, especialmente a mi madre, María Loayza; mi padre, David Aguirre, y a mis hermanos, quienes con su sacrificio y apoyo me dieron aliento para realizar mis estudios.

A mi novia, quien me ayudo con todo lo que estuvo a su alcance.

A mi tío, Richard Apolo, quien me apoyo con su conocimiento y materiales para realizar mis proyectos a lo largo de la carrera.

Al Ph.D. Marcelo Fajardo y al M.Sc. Efraín Terán, su confianza y orientación fueron fundamentales para el desarrollo del proyecto. Al Ph.D Carlos Saldarriaga, su apoyo en este último semestre me dio la confianza para enfrentar los desafíos que se presentaron.

André Aguirre

Agradecimientos

Gracias a Dios, que me ha dado salud y sabiduría durante todo mi camino universitario. A mi familia, por las palabras de aliento que me brindaron en los momentos más difíciles. Y a mis tutores, por todo el tiempo dedicado para que este proyecto se cumpla.

Alexandra Imaicela

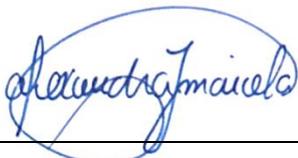
Declaración Expresa

Nosotros André Aguirre Apolo y Alexandra Imaicela Cañizares 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 a 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, 26 de enero del 2024.



Alexandra Imaicela Cañizares



André Aguirre Apolo

Evaluadores

Efraín Terán Calle, M.Sc.

Profesor de materia

Marcelo Fajardo Pruna, Ph.D.

Tutor de proyecto

Resumen

En este mundo donde la tecnología crece exponencialmente, los avances en los temas de realidad virtual y simuladores han permitido crear entornos virtuales interactivos. Uno de los tipos de simuladores de interés son los de navegación, imprescindibles para la formación de estudiantes aspirantes a marinos para la Fuerza Naval.

En este proyecto se diseña un entorno físico de simulación para la cabina de una fragata naval, mediante un enfoque mecatrónico que optimice la preparación y competencia de los timoneros en flotas marinas; mostrando que los simuladores de navegación resultan ser sistemas ideales para el aprendizaje de maniobras de un entorno controlado y seguro. El proyecto surge como respuesta a costosos simuladores, exportación de datos privados y pérdidas de tiempo en actualizaciones y renovaciones de software.

Para realizar este proyecto, se utilizó Unity para el diseño del entorno virtual y la placa Arduino Mega para la transmisión de datos de movimiento al buque. El resultado obtenido fue un entorno de simulación, junto con un prototipo de consola. Se concluyó que es posible crear un entorno virtual marítimo en el que un navío es controlado mediante actuadores físicos. Los futuros avances del proyecto son esenciales para garantizar un mejor entrenamiento del personal.

Palabras clave: simulador de navegación, entrenamiento naval, realidad virtual, entorno virtual marítimo

Abstract

In this world where technology grows exponentially, advances in virtual reality and simulators have made it possible to create interactive virtual environments. One of the types of simulators of interest are navigation simulators, essential for the training of aspiring sailor students for the Naval Force.

In this project, a physical simulation environment is designed for the cockpit of a naval frigate, using a mechatronic approach that optimizes the preparation and competence of helmsmen in marine fleets; showing that navigation simulators turn out to be ideal systems for learning maneuvers in a controlled and safe environment. The project arises in response to expensive simulators, export of private data and wasted time on software updates and renewals.

To carry out this project, Unity was used for the design of the virtual environment and the Arduino Mega board for the transmission of movement data to the ship. The result obtained was a simulation environment, along with a console prototype. It was concluded that it is possible to create a maritime virtual environment in which a ship is controlled by physical actuators. Future progress of the project is essential to guarantee better staff training.

Keywords: navigation simulator, naval training, virtual reality, maritime virtual environment.

Abreviaturas

ASME	American Society of Naval Engineers
ATMEL	Advanced Technology Microcontrollers
CIMAR	Centro de Instrucción y Capacitación Marítima
DAQ	Data Acquisition
IDE	Entorno de Desarrollo Integrado
NMI	Northeast Maritime Institute
VBS	Virtual Battlespace
ESPOL	Escuela superior politécnica del litoral
USD	United states dollar
3D	Tres dimensiones
GND	Ground
IDE	Integrated Development Environment
DC	Corriente directa
SRAM	Static Random Access Memory
EEPROM	Electrically Erasable Programmable Read-Only Memory
USB	Universal Serial Bus
PWM	Pulse Width Modulated
LED	Light-emitting diode
UART	Universal Asynchronous Receiver-Transmitter
SPI	Serial Peripheral Interface
HD	High definition

Simbología

mm	Milímetro
cm	Centímetro
m	Metro
mV	Milivoltios
V	Voltio
mA	Miliamperios
H	Hertzios
MH	MegaHertzios
mH	miliHertzios
A	Amperio
B	Byte
TB	Terabyte
KB	Kilobyte
GB	Gigabyte
tons	Toneladas
in	Pulgada
Ω	Ohmio
k Ω	Kilo-ohmio

Índice general

Resumen	I
Abstract.....	II
Abreviaturas.....	III
Simbología.....	IV
Índice general.....	V
Índice de figuras	VIII
Índice de tablas	X
ÍNDICE DE PLANOS.....	XI
Capítulo 1	1
1.1 Descripción del problema.....	4
1.2 Justificación del problema.....	6
1.3 Objetivos.....	7
1.3.1 Objetivo general	7
1.3.2 Objetivos específicos.....	7
1.4 Marco teórico.....	8
1.4.1 Entorno de simulación	8
1.4.2 Sistemas de adquisición de datos (DAQ)	9
1.4.3 Entornos virtuales	10

1.4.4	Integración de entorno virtual con entradas y salidas físicas	12
1.4.5	Placa electrónica Arduino.....	13
1.5	Estado del arte	15
Capítulo 2	18
2.1	Requerimientos y limitaciones	19
2.2	Selección de alternativa de solución.....	21
2.3	Proceso de diseño	25
2.4	Diseño conceptual.....	27
2.5	Parámetros de diseño electrónico	29
2.5.1	Selección de placa electrónica.....	29
2.5.2	Selección de componentes electrónicos.....	31
2.5.3	Selección de fuente de poder	39
2.5.4	Diagrama eléctrico.....	40
2.5.5	Adaptación de señales.....	41
2.6	Parámetros de diseño mecánico.....	43
2.6.1	Diseño de embarcación.....	43
2.6.2	Diseño de la consola de mando	47
2.6.3	Plano de implantación	48
2.7	Parámetros de diseño de programación.....	50
2.7.1	Selección de Entorno de Desarrollo Integrado (IDE).....	50

2.7.2	Diseño del entorno virtual	54
2.7.3	Algoritmos de navegación y condiciones de manejo.	55
Capítulo 3	57
3.1	Diseño de consola de mando	58
3.2	Diseño del entorno virtual	59
3.3	Interacción del entorno con actuadores físicos.....	59
3.4	Consumo energético	61
3.5	Costos	63
3.5.1	Costos de equipos	63
3.5.2	Costos de implementación.....	63
Capítulo 4	66
4.1	Conclusiones.....	67
4.2	Recomendaciones	68
Referencias	70
Apéndices	79

Índice de figuras

Figura 1.1 Demostración de simulación contra incendios.....	3
Figura 1.2 Simulador de navegación MissionQuest - Fast rescue craft training.....	4
Figura 1.3 Buques navieros Fragata Moran Valverde y tanquero Atahualpa.....	5
Figura 1.4 Simulador STAVE-LCS.....	9
Figura 1.5 Flight simulator de la Armada Estadounidense	10
Figura 1.6 Nautis Simulator.....	11
Figura 1.7 Partes generales del Arduino MEGA.....	14
Figura 1.8 Entorno de Desarrollo Integrado programado con Arduino Code	15
Figura 1.9 Simulador de aeronave del departamento de defensa estadounidense.	16
Figura 1.10 Entorno de simulación de VSTEP.....	17
Figura 2.1 Alternativas de solución de proyecto	21
Figura 2.2 Proceso de diseño para un sistema de entorno virtual de navegación.....	25
Figura 2.3 Diseño conceptual de la solución	27
Figura 2.4 Diagrama de secuencia de la solución.....	28
Figura 2.5 Diagrama esquemático Arduino ATMEGA2560.....	31
Figura 2.6 Fuente de poder seleccionada.....	40
Figura 2.7 Diagrama de conexiones del proyecto	41
Figura 2.8 Módulo convertidor de corriente a voltaje	42
Figura 2.9 Dibujo a escala de una fragata tipo Leander	44
Figura 2.10 Modelo réplica de la embarcación FRAGATA	47
Figura 2.11 Diseño de la consola de mando	48
Figura 2.12 Diseño de distribución del simulador de navegación y la consola.....	49

Figura 2.13 Plataforma de Microsoft Visual Studio enlazado a Unity	52
Figura 2.14 Software de Arduino	53
Figura 2.15 Pantalla principal de Unity	54
Figura 2.16 Imagen del ambiente virtual de Unity	55
Figura 3.1 Embarcación de baja resolución.....	60

Índice de tablas

Tabla 2.1 Requerimientos para el desarrollo del proyecto	20
Tabla 2.2 Criterios de valoración.....	22
Tabla 2.3 Evaluación del peso específico de cada criterio de valoración	23
Tabla 2.4 Selección de solución del proyecto	24
Tabla 2.5 Especificaciones del Arduino MEGA	30
Tabla 2.6 Característica de un selector 2 posiciones	32
Tabla 2.7 Característica de un selector 3 posiciones	33
Tabla 2.8 Característica de la palanca de mando NORISYS 4 LS4	34
Tabla 2.9 Características del joystick de la consola de mando.....	35
Tabla 2.10 Volante con potenciómetro.....	36
Tabla 2.11 Televisor TCL 50S450G	37
Tabla 2.12 Monitor para consola de mando AOC 20E1H HD.....	38
Tabla 2.13 Características recomendadas de CPU	39
Tabla 2.14 Características generales del buque fragata.....	45
Tabla 2.15 Dimensiones y peso del buque fragata	46
Tabla 2.16 Capacidades del buque fragata	46
Tabla 2.17 Diferencias entre plataformas de programación.....	51
Tabla 3.1 Detalles del consumo energético	61
Tabla 3.2 Detalles de costo de componentes.....	62
Tabla 3.3 Detalles de costos de ingeniería.....	64

ÍNDICE DE PLANOS

Plano 1. Diseño de consola de mando.	100
Plano 2. Medidas del tablero de la consola de mando.....	101
Plano 3. Medidas del plano de implantación de la consola de man	102

Capítulo 1

1 Introducción

Los simuladores constituyen a nivel mundial uno de los sistemas más importantes para la capacitación del personal militar. Estos sistemas logran representar, de forma interactiva, situaciones que el personal podría enfrentar en el mundo real. El uso de simuladores permite que el conocimiento teórico sea complementado con clases prácticas dentro de un ambiente seguro; brindando la experiencia necesaria para que el personal desempeñe su papel sin exponerse a situaciones de riesgo real.

Con el auge de la tecnología, los simuladores y la realidad virtual se han consolidado como una base importante para formar personal civil y militar. Los nuevos estudiantes, con el uso de estos sistemas, han llegado a manejar diferentes tipos de vehículos: aéreos, terrestres y marinos [1]. Esta herramienta tecnológica ha potenciado la experiencia y el entrenamiento de los conductores en los vehículos tripulados; dado que, el escenario es controlable y configurable de acuerdo con las variables que el instructor coloque. A más de ello, la experiencia realista que experimenta el sujeto durante la práctica facilita evidenciar el comportamiento que manejará ante escenarios imprevistos o situaciones de fallas en el transporte [2].

Figura 1.1

Demostración de simulación contra incendios



Nota. La figura muestra un simulador para bomberos, donde se está recreando una escena con el uso de gafas de realidad virtual y un controlador (manguera contra incendios) [3].

En cuanto a la formación naval, el mar ha se ha convertido en un verdadero desafío para estas instituciones. Las condiciones climáticas, los peligros y costes que representa una práctica en alta mar; motivan el uso de sistemas de simulación de navegación en el puente de gobierno, para los guardiamarinas y todo el personal militar que necesite preparación en esta área. Además, debido a la información sensible que se maneja dentro de instituciones navales, la mayoría de los gobiernos ha optado por crear sus propios simuladores [3].

Figura 1.2

Simulador de navegación MissionQuest - Fast rescue craft training



Nota. La figura muestra un tipo de simulador de navegación, en el que estudiantes se encuentran practicando un tipo de escenario con condiciones preestablecidas [4].

1.1 Descripción del problema

En el contexto de la seguridad marítima nacional, es crucial que el personal militar se encuentre capacitado para operar los diversos navíos de manera eficiente, segura y óptima.

El problema que aborda este proyecto proviene de la necesidad de proporcionar un entorno de entrenamiento simulado propio que sea efectivo y realista, en la que su uso sea ilimitado y sin restricciones.

Las simulaciones proporcionan un ambiente seguro y controlado en el que el personal marino puede desarrollar habilidades como: navegación, toma de decisiones, familiarización con el funcionamiento de buques y escases de errores en las operaciones de estos. La importancia de preparar y adiestrar al personal militar radica en las situaciones críticas que se dan en el mar; donde, la seguridad, agilidad y efectividad son prioritarias; por ello, el simulador estará orientado a la preparación de los estudiantes guardiamarinas y oficiales para el manejo del navío Fragata; un ejemplo de la estructura de este buque se lo puede apreciar en la Figura 1.3.

Figura 1.3

Buques navieros Fragata Moran Valverde y tanquero Atahualpa



Nota. En la fotografía se puede apreciar dos tipos diferentes de buques los cuales son utilizados en la marina ecuatoriana [5].

1.2 Justificación del problema

Los elevados costos de simuladores de pago, el manejo de datos privados del país y las demoras en la actualización y renovación del software, generan pérdidas de tiempo para los cadetes navales y la dependencia de la tecnología externa. Estos factores han provocado que los nuevos tripulantes no se encuentren preparados a tiempo completo, disminuyendo su experiencia de maniobrar en una variedad de entornos.

Actualmente, transferir a oficiales de la marina a otros países para que se capaciten y realicen sus prácticas de manera real, supone un costo elevado por cada misión. Por ello, este proyecto es de importancia estratégica debido a que contribuye a la: seguridad marítima nacional, defensa de fronteras e inversión económica a largo plazo.

La preparación efectiva del personal militar es esencial para proteger los intereses marinos y mantener la soberanía en aguas nacionales. La rentabilidad que ofrece el sistema de simulación es sumamente alta, ya que no se necesita ejecutar una salida en un navío real; disminuyendo costos y tiempo de espera de las distintas prácticas que deban realizar los tripulantes.

Tiempo atrás, el Ecuador presentaba limitaciones en su tecnología, tanto en la logística como en su personal capacitado. Hoy se posee muchos recursos que permiten crear un sistema propio de realidad virtual para entrenar en la navegación.

1.3 Objetivos

1.3.1 *Objetivo general*

Diseñar un entorno físico de simulación para la cabina de una fragata naval mediante un enfoque mecatrónico con el fin de optimizar la preparación y competencia de los timoneros en flotas marinas.

1.3.2 *Objetivos específicos*

- Definir las entradas y salidas requeridas por el entorno de simulación, que cumpla con los requerimientos de funcionalidad establecidos por el cliente.
- Desarrollar un entorno virtual funcional mediante el uso de programas computacionales de código abierto.
- Seleccionar los elementos de interacción física necesarios para la navegación en el entorno virtual desarrollado.
- Especificar el plano de implantación del sistema, en el que se muestre una adecuada ubicación de los componentes físicos del entorno virtual, a fin de evitar interferencia durante el uso del simulador.
- Presentar la documentación técnica de las diferentes ingenierías que permita validar y replicar el sistema en futuros proyectos.

1.4 Marco teórico

La simulación en el entrenamiento militar es muy utilizada en todo el mundo. Se ha demostrado que proporciona un ambiente de entrenamiento seguro y realista para las fuerzas armadas; permitiendo a los militares adquirir habilidades y experiencia sin exponerse a riesgos innecesarios. Investigaciones internacionales han destacado la eficacia de la simulación en la formación del personal militar, además, contribuye a la mejora de la preparación y la seguridad en operaciones reales [6].

1.4.1 Entorno de simulación

La creación de entornos de simulación física para la capacitación militar es un campo de estudio bien establecido. Investigaciones previas han abordado la replicación precisa de cabinas y sistemas de control de buques de guerra. Estos entornos físicos simulados han demostrado ser esenciales para la inmersión y el entrenamiento efectivo de las tripulaciones.

Proyectos como el simulador Surface Training Advanced Virtual Environment–Littoral Combat Ship (STAVE-LCS) de la marina estadounidense, han desarrollado entornos de simulación física altamente precisos que replican las cabinas de buques de guerra. Estos entornos permiten una inmersión completa y efectiva para la formación de tripulaciones de fragatas [7].

Figura 1.4

Simulador STAVE-LCS



Nota. Captura del video del simulador STAVE-LCS [8].

1.4.2 Sistemas de adquisición de datos (DAQ)

El uso de sistemas de adquisición de datos para capturar las entradas de los controles de cabina es una práctica común en la simulación. La literatura técnica internacional ha explorado ampliamente la integración de DAQ en entornos de simulación y su importancia para la precisión y la sincronización de las entradas.

En la industria de la simulación, el *Link-11/22 Simulation System* se basa en sistemas de adquisición de datos para capturar entradas de control. Este sistema es esencial para garantizar la precisión y sincronización de las entradas en entornos de simulación complejos [9].

Figura 1.5

Flight simulator de la Armada Estadounidense



Nota. La imagen muestra el uso por parte del personal militar del simulador de vuelo de la armada estadounidense basado en sistemas de adquisición de datos [10].

1.4.3 Entornos virtuales

La creación de entornos virtuales utilizando plataformas como Unity, Unreal Engine, entre otros; es una tendencia actual en la simulación militar. Investigaciones previas han demostrado que estos entornos ofrecen una experiencia de entrenamiento altamente inmersiva y realista, proporcionando al ejército una interacción con modelos virtuales de equipos y escenarios militares. El VirtualShip desarrollado por General Dynamics es un ejemplo destacado de la creación de entornos virtuales utilizando Unity. Este sistema proporciona a los militares una experiencia altamente inmersiva para interactuar con modelos virtuales [11].

Las tecnologías de realidad virtual permiten a los soldados experimentar cualquier vehículo de forma más realista que usando el *mouse* y teclado o un timón de simulador de conducción.

Creados por científicos, los simuladores de vehículos de realidad virtual militares constituyen una plataforma compleja que recrea aspectos como la forma y la sensación del vehículo, su maniobrabilidad en varios tipos de terreno y su comportamiento en diferentes condiciones climáticas. Además de eso, un soldado puede navegar en entornos nativos de cualquier parte del mundo, incluidos asentamientos reconstruidos de cualquier tipo [12].

Figura 1.6

Nautis Simulator



Nota. Entorno de simulación de Nautis Simulator, donde observan las diferentes pantallas con el entorno virtual, así como las entradas del sistema [13].

1.4.4 Integración de entorno virtual con entradas y salidas físicas

Los sistemas de entrada física, que incluyen dispositivos como controles, botones, palancas, joysticks, pedales, entre otros; son manipulados por el usuario para interactuar con el simulador. Estos dispositivos representan los controles y operaciones reales del entorno que se está simulando como, el timón de un barco, el joystick de un avión o los controles de un vehículo [14].

Los sistemas de adquisición de datos se encargan de capturar las señales generadas por los dispositivos de entrada física. Utilizan sensores, convertidores analógico-digitales (ADC), interfaces y otros dispositivos para traducir las entradas físicas en datos digitales que el sistema pueda comprender y procesar.

Los datos adquiridos se envían al sistema de procesamiento y modelado. Aquí, los datos se interpretan para aplicarlos a modelos matemáticos y al entorno virtual llegando a simular el comportamiento del sistema real. Por ejemplo, las acciones del usuario en el timón se traducen en cambios en la dirección y la velocidad de un barco en el entorno virtual [15].

El entorno virtual es la representación digital del entorno físico que se está simulando. Incluye gráficos 3D, escenarios, vehículos, instrumentos y otros elementos necesarios para recrear la experiencia del usuario. Las acciones y comandos generados por el usuario se reflejan en este entorno virtual, modificando la visualización o interacciones dentro del simulador [15].

Las respuestas del entorno virtual se traducen de nuevo en señales físicas a través de sistemas de salida. Estos sistemas pueden incluir monitores, vibraciones, actuadores o cualquier dispositivo que permita al usuario recibir retroalimentación sensorial de las acciones realizadas en el entorno virtual [16].

Finalmente, se produce la retroalimentación. Facilita información al usuario sobre las acciones realizadas y cómo afectan al entorno simulado. Esto cierra el ciclo al permitir que el usuario ajuste su comportamiento en función de la respuesta recibida, facilitando un entrenamiento más efectivo [16].

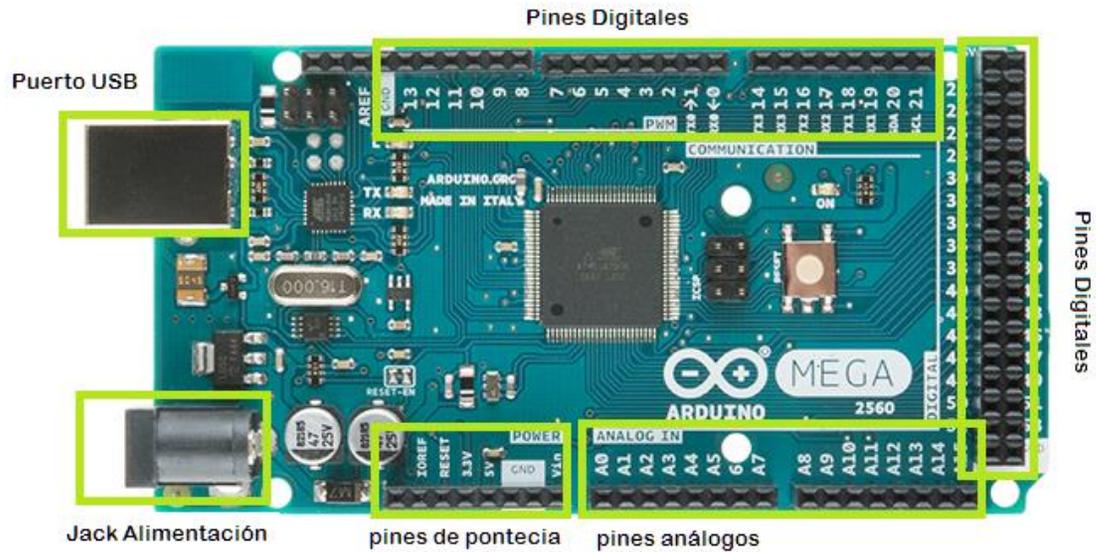
1.4.5 Placa electrónica Arduino

En el mundo de la electrónica el término Arduino es conocido como una empresa que se dedica al desarrollo de *hardware* y *software* de código abierto. El público de interés de esta empresa son personas dedicadas a la construcción de proyectos relacionados a la electrónica y la informática.

Las placas de circuito impreso que fabrica la empresa Arduino integran un microcontrolador ATMEL (Advanced Technology Microcontrollers) que cuenta con: pines (GND, alimentación, analógico y digitales), un *plug* de alimentación y una entrada USB. Esta placa se encuentra conformado de una unidad central de procesamiento (microprocesador), una memoria y periféricos. El microcontrolador es un circuito integrado programable que se encarga de ejecutar y grabar instrucciones específicas de acuerdo con el control de las E/S (entradas/salidas) y la gestión de interrupciones, proporcionando una relación entre el usuario y el sistema [17].

Figura 1.7

Partes generales del Arduino MEGA

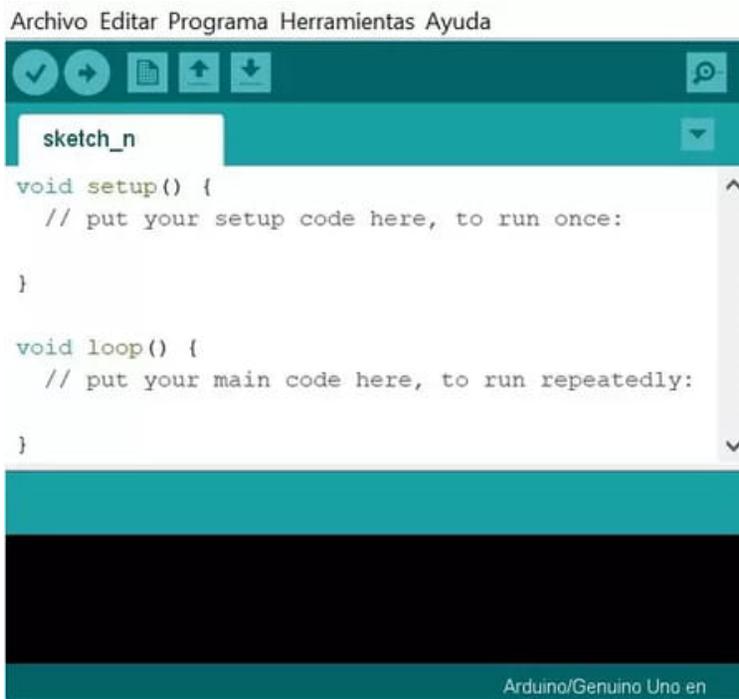


Nota. Se presenta en la figura la partes que compone el *hardware* del ArduinoATMEGA2560 [18].

El *software* de Arduino cuenta con su propio IDE para el desarrollo del código de programación. Esta plataforma se caracteriza por ser; libre, accesible, flexible y fácil de usar. El IDE de Arduino se lo programa con su propio tipo de lenguaje llamado *Arduino code*; este tipo de lenguaje de programación se encuentra basado en C++ e integra una gran variedad de funciones y métodos que son específicas del entorno.

Figura 1.8

Entorno de Desarrollo Integrado programado con Arduino Code



The image shows a screenshot of the Arduino IDE interface. At the top, there is a menu bar with 'Archivo', 'Editar', 'Programa', 'Herramientas', and 'Ayuda'. Below the menu bar is a toolbar with icons for saving, running, uploading, and downloading. The main window displays a sketch named 'sketch_n' with the following code:

```
void setup() {  
  // put your setup code here, to run once:  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  
}
```

At the bottom of the IDE, there is a status bar that reads 'Arduino/Genuino Uno en'.

Nota. Se muestra dentro del IDE la programación de Arduino *code* que aparece de forma predeterminada al iniciar un nuevo proyecto [17].

1.5 Estado del arte

A nivel global, los avances sobresalientes en la implementación de tecnologías de simulación y sistemas de entrenamiento para las fuerzas armadas en todo el mundo son notables. Numerosos países han liderado el camino en el desarrollo de simuladores de alto rendimiento, capaces de adiestrar a sus tripulaciones en una amplia gama de situaciones, desde operaciones de combate hasta tareas de navegación crítica. Ejemplos notables incluyen centros de simulación y capacitación en Estados Unidos, donde se han incorporado simuladores de fragatas, submarinos y aeronaves de última generación.

Figura 1.9

Simulador de aeronave del departamento de defensa estadounidense.



Nota. La imagen muestra un simulador de aeronave del departamento de defensa estadounidense creado para el entrenamiento de pilotos de combate [19].

En cuanto a sistemas de adquisición de datos (DAQ), se lograron avances significativos a nivel global, con empresas líderes como National Instruments ofreciendo soluciones de vanguardia utilizadas en varias aplicaciones, incluyendo simulación y entrenamiento militar. La capacidad de capturar datos precisos y sincronizados de los controles de la cabina es fundamental para crear experiencias de entrenamiento realistas y efectivas. En cuanto a entornos virtuales, se han desarrollado proyectos de alta tecnología en universidades y empresas en todo el mundo. El *Virtual Battlespace* (VBS) de *Bohemia Interactive Simulations* o VSTEP son un ejemplo destacado, brindando a los tripulantes una experiencia altamente inmersiva para interactuar con modelos virtuales de equipos y escenarios complejos.

Figura 1.10

Entorno de simulación de VSTEP



Nota. La imagen muestra la cabina de un buque carguero, este simulador es utilizado para aprender maniobras en el mar [13].

Capítulo 2

2 Metodología

En este apartado se muestra las etapas del proceso de diseño, tomando en cuenta los requerimientos del sistema y los objetivos planteados. Se abarca temas desde los requerimientos solicitados por el cliente; alternativas de solución; parámetros del diseño desde la perspectiva electrónica, mecánica y computacional; hasta las limitaciones de la solución propuesta.

2.1 Requerimientos y limitaciones

En el sistema de simulación de navegación, se establecieron diversos tipos de requerimientos dependiendo de las necesidades del cliente. La Tabla 2.1 presenta dichas directrices.

La limitación de este proyecto se presentó en los componentes que se utilizaron para verificar el funcionamiento. El cliente al ser una gran institución requiere de equipos de grado industrial y robustos en tecnología; sin embargo, en el prototipo de baja resolución que se realizó, se implementaron componentes electrónicos que son similares en su funcionamiento, a más de ser accesibles y económicos. Las limitaciones prácticamente se refieren a los costos de los componentes, existencia de los componentes electrónicos dentro del mercado ecuatoriano, accesibilidad de *softwares* que sean de código abierto y el tiempo de programación de programas. Las limitaciones también se presentaron en la accesibilidad de la información, al ser una institución miliar no fue sencillo tener acceso a ciertos planos e información necesaria para el desarrollo, principalmente en el apartado de diseño mecánico de la embarcación y diseño del plano de implantación.

Tabla 2.1*Requerimientos para el desarrollo del proyecto*

Requerimiento	Descripción
Computacional	<ul style="list-style-type: none"> • Interfaz de diseño ambientado a un entorno marino. • Entorno virtual con control de movimiento del naviero dependiendo de sus especificaciones. • Procesamiento de señales analógicas y digitales con su respectiva interpretación en el entorno virtual.
Electrónico	<ul style="list-style-type: none"> • Microcontrolador compatible con el <i>software</i>. • Selector dos posiciones para el mando del telemotor. • Selectores tres posiciones para el modo de selección de bombas. • Primer modo de control: Potenciómetro que simule al timón. • Segundo modo de control: Joystick para el movimiento del barco virtual.
Control	<ul style="list-style-type: none"> • Movimiento rotacional, izquierdo y derecho, entorno al eje Y. • Movimiento lineal, adelante y atrás, entorno al plano XZ. • Codificación de las diferentes condiciones de manejo

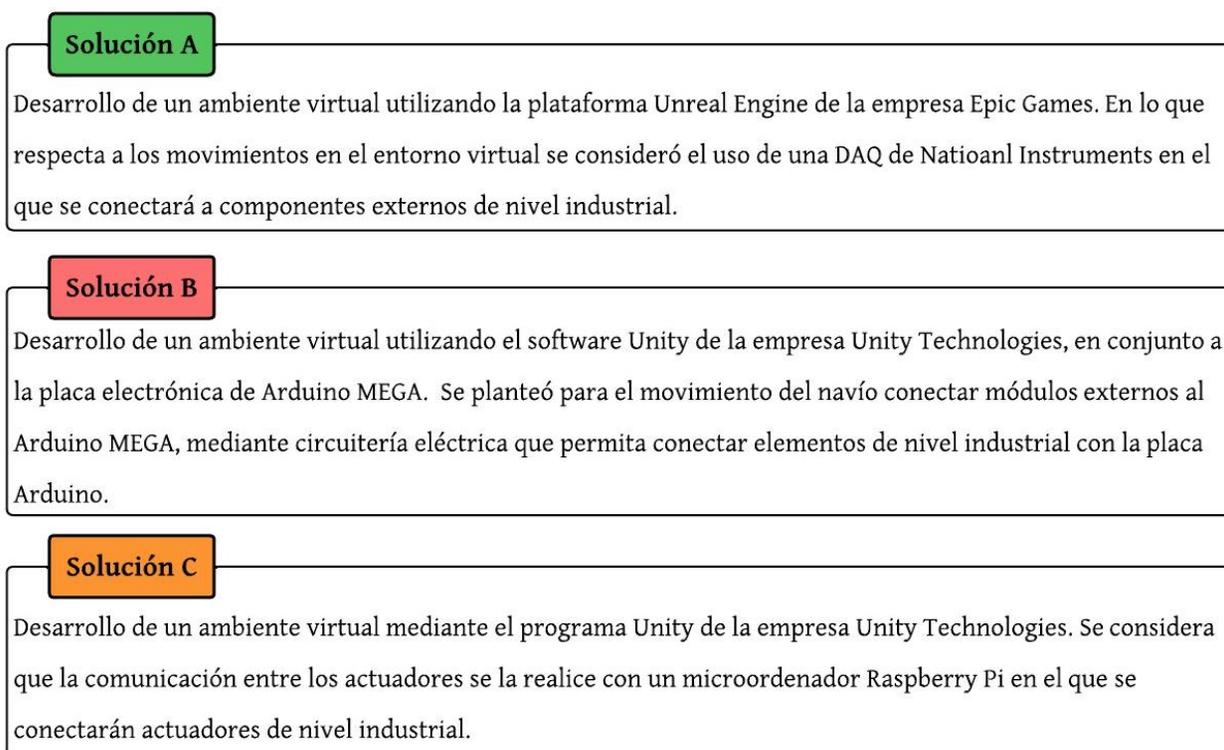
Nota. Los requerimientos se determinaron en base a investigación previa de simuladores existentes y análisis de funcionamiento de cada una.

2.2 Selección de alternativa de solución

Analizando el problema definido en el proyecto, se plantearon diversas alternativas de solución. Entre las alternativas consideradas, resaltaron tres tipos de solución, las cuales se muestran en la Figura 2.1.

Figura 2.1

Alternativas de solución de proyecto



Para la selección de la alternativa más viable, se planteó criterios de valoración de diseño de acuerdo con las especificaciones establecidas por el cliente. Los criterios de valoración resultan ser aspectos generales que se esperan del proyecto, y se ordena cada uno desde el más importante al

menos importante. En la Tabla 2.2 se detalla las características que se consideraron para seleccionar una solución final entre las tres alternativas propuestas.

Tabla 2.2

Criterios de valoración

Criterio	Descripción
Actualización	Modificaciones en el software del simulador para la corrección de errores y mejora continua de experiencias.
Calidad de gráficas	Visibilidad, sombras y dimensionamiento de los objetos que se muestran en el programa.
Costos	Cotización de todos los materiales a utilizar en el proyecto.
Estética	Contraste de colores y diseño de objetos visualizados en el simulador.
Facilidad de uso	Interfaz del simulador intuitivo en conjunto a los actuadores manuales.
Precisión	Movimientos del barco acorde a las interacciones con los actuadores con un error del 2%.
Fiabilidad del software	Software que permita el desarrollo autónomo para evitar salida de datos hacia entes externos.
Viabilidad	Uso de software sin limitaciones en los IDE y libertad de uso de librerías.

Nota. Los criterios seleccionados se consideraron al revisar las alternativas planteadas.

Por medio de una matriz de priorización, se determinó el peso específico de cada uno de los criterios. Para ello, se establecieron las siguientes ponderaciones:

- **1**; Cuando el criterio de solución de las filas es superior que el de las columnas.
- **0.5**; Cuando el criterio de solución de las filas es igual de importante que el de las columnas.
- **0**; Cuando el criterio de solución de las filas es peor que el de las columnas.

La Tabla 2.3 muestra en detalle las ponderaciones de cada uno de los criterios.

Tabla 2.3

Evaluación del peso específico de cada criterio de valoración

Criterio	Viabilidad	Costos	Actualización	Precisión	Facilidad de uso	Calidad de gráficas	Fiabilidad de soft.	Estética	$\sum +1$	Ponderación (%)
Viabilidad	-	1	0.50	1	1	1	1	1	7.50	20.83
Costos	0	-	1	1	1	1	1	1	7.00	19.44
Actualización	0.50	0	-	1	1	1	1	1	6.50	18.06
Precisión	0	0	0	-	0.50	1	1	1	4.50	12.50
Facilidad de uso	0	0	0	0.50	-	1	1	1	4.50	12.50
Calidad de gráficas	0	0	0	0	0	-	0.50	1	2.50	6.94
Fiabilidad de soft.	0	0	0	0	0	0.50	-	1	2.50	6.94
Estética	0	0	0	0	0	0	0	-	1.00	2.79
Total									36	100

Nota. Las ponderaciones se basaron en los requerimientos del cliente.

Viabilidad > Costos > Actualizaciones > Precisión = Facilidad de uso
> Calidad de gráficas = Fiabilidad de software > Estética

La selección de la alternativa se realizó con una matriz de priorización entre filas y columnas. Se analizaron los criterios de cada una de las soluciones y se escogió aquella con el mayor peso. En la siguiente tabla se muestra el proceso de selección:

Tabla 2.4

Selección de solución del proyecto

Ponderación (%)	20.83	19.44	18.06	12.50	12.50	6.94	6.94	2.79		
Conclusión	Viabilidad	Costos	Actualización	Precisión	Facilidad de uso	Calidad de gráficas	Fiabilidad de soft.	Estética	Peso promedio	Prioridad
Solución A	0.50	0.50	0	0	0	0.50	0	0	0.24	2
Solución B	0.50	0.50	1	0	1	0.50	0	1	0.57	1
Solución C	0	0	0	1	0	0	1	0	0.19	3
								Total	1	

Nota. Las ponderaciones de los criterios se obtuvieron de la Tabla 2.3

De la Tabla 2.4 se evidencia que la solución B es la más adecuada para implementar en el proyecto; debido a que, se alinea a los requerimientos solicitados por el cliente y a su vez cumple con los criterios de valoración establecidos en la Tabla 2.2.

2.3 Proceso de diseño

La Figura 2.2 muestra las diferentes etapas del proceso de diseño que permiten cumplir los objetivos propuestos en el capítulo 1. El proceso de diseño de este proyecto combina diferentes tipos de disciplinas como: programación, mecánica y electrónica. En cada una de estas etapas del proceso de diseño se describe la secuencia a seguir, desde el inicio del proceso que empieza con una reunión con el cliente hasta la obtención del producto o prototipo final esperado.

Figura 2.2

Proceso de diseño para un sistema de entorno virtual de navegación



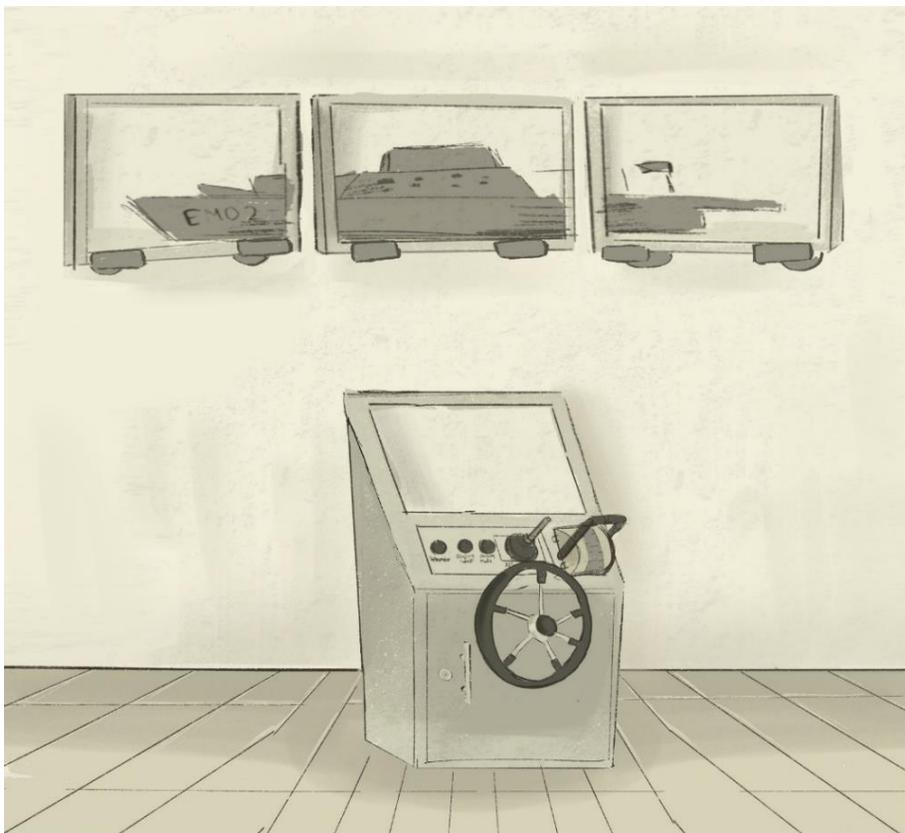
Nota. El diagrama de proceso de diseño se elaboró en la plataforma Lucid-chart.

2.4 Diseño conceptual

En la Figura 2.3 se muestra un boceto general de la solución propuesta a la problemática, en donde se observan los principales componentes que forman parte del simulador. Entre los elementos se identifican: los monitores, donde se presentará el entorno virtual desarrollado; el monitor principal, donde el usuario empezará la simulación; los selectores de tres y dos posiciones, que permiten elegir entre distintas configuraciones de manejo; el joystick de control, el timón del barco y un controlador diferencial.

Figura 2.3

Diseño conceptual de la solución

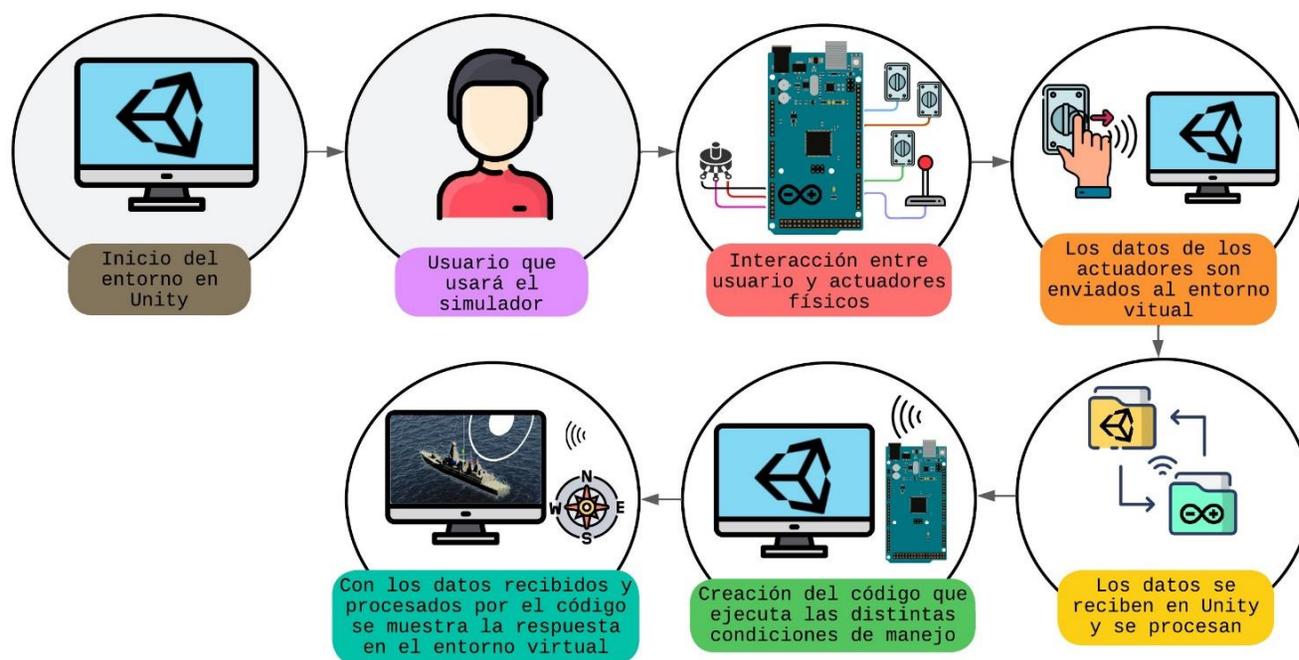


Nota. Diseño conceptual Realizado en Infinite Painter.

El diseño conceptual consistió en crear un entorno simulado de navegación haciendo uso de herramientas informáticas y electrónicas. El entorno virtual en Unity se desarrolló con el editor de código Microsoft Visual Studio y la interfaz gráfica del mismo software de Unity. En la Figura 2.4 se muestra el proceso realizado para transmitir datos de movimiento entre los componentes electrónicos conectados al Arduino y el programa Unity; para la transferencia de datos se realizó con el IDE de Arduino.

Figura 2.4

Diagrama de secuencia de la solución



Nota. Diagrama elaborado en Lucidchart.

2.5 Parámetros de diseño electrónico

En este apartado, se presenta la selección de la placa electrónica para la comunicación entre el software de Unity y los componentes de nivel industrial que se le recomienda al cliente que utilice.

2.5.1 Selección de placa electrónica

El microcontrolador ATMEGA2560 R3 es uno de los diversos tipos de placas integradas que permiten el control de entradas digitales y analógicas haciendo uso de la programación de Arduino *code*. Se seleccionó este tipo de placa debido a que posee una alta rapidez de procesamiento y una gran cantidad de pines de entrada. En la Tabla 2.5 se presentan las especificaciones técnicas del controlador seleccionado que demuestra su eficiencia para proyectos de tipo robusto.

La Figura 2.5 muestra el microcontrolador ATMEGA2560. Se planeó usar una tarjeta de adquisición de datos de National Instruments; sin embargo, el costo de esta tarjeta, el software de paga utilizado para acceder a la misma y el uso de programas con propietarios que pueden presentar restricciones sobre su uso y modificación, dio lugar para desistir del uso de este equipo y optar por el uso de un software libre. Concluyendo que, el ATMEGA2560 es el componente adecuado para emplearse junto al entorno virtual de Unity.

Tabla 2.5*Especificaciones del Arduino MEGA*

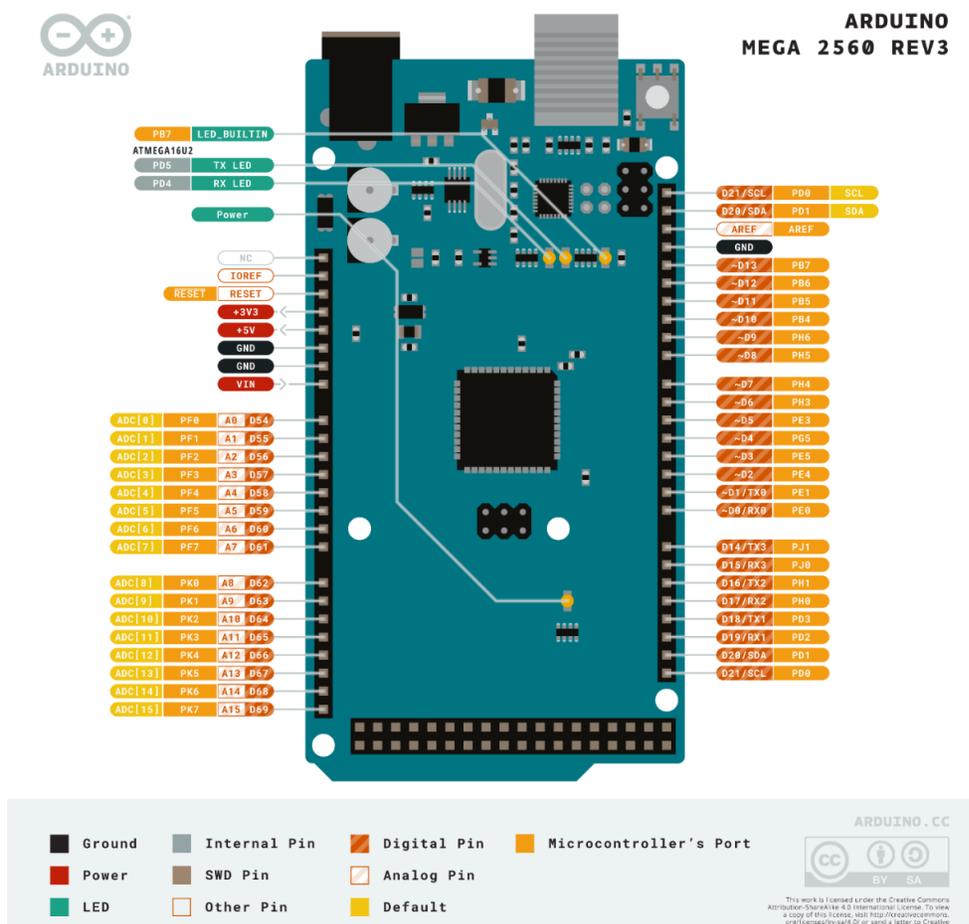
Componentes	Características
Microcontrolador	ATmega2560
Pines	54 digitales, 16 analógicas, 15 PWM y 13 LED incorporado
Comunicación	UART, I2C y SPI
Poder	El voltaje nominal: 7-12 [V]
	Corriente continua por pin de E/S: 20 [mA] Conector tipo tapón de barril
Velocidad de reloj	Procesador principal ATmega2560 de 16 [MHz] y procesador USB-serie ATmega16U2 de 16 [MHz]
Memoria	ATmega2560 de 28 [KB] de SRAM, 256 [KB] FLASH y 4 [KB] de EEPROM

Nota. Datos obtenidos de la página oficial de la compañía Arduino [20].

Cabe destacar que el ATMEGA2560 posee gran cantidad de pines analógicos y digitales; una mayor cantidad de memoria SRAM, FLASH y EEPROM; y otro microcontrolador que mejora la potencia de procesamiento.

Figura 2.5

Diagrama esquemático Arduino ATMEGA2560



Nota. Obtenido de la ficha técnica de Arduino [20].

2.5.2 Selección de componentes electrónicos

En este apartado se presenta los componentes seleccionados para interactuar con la placa electrónica y permite la captación de señales analógicas y digitales que controlan el movimiento del entorno de simulación en Unity. Para el movimiento se seleccionaron opciones que emulen el entorno en que el timonero estaría al momento de realizar una misión real. Además, la elección se

basa en la ventaja de que la mayoría de los actuadores se pueden adquirir localmente con respaldo y garantía de las empresas que las distribuye.

Tabla 2.6

Figura	Características
	<p>N° de referencia: 1NA 3SU11502BF601BA0</p> <p>Marca: Siemens</p> <p>Voltaje (Regulable): 5-500 [V_{DC}]</p> <p>Corriente: 10 [A]</p>

Característica de un selector 2 posiciones

Nota. Datos obtenidos de la ficha técnica del componente [21].

La Tabla 2.14 muestra el selector seleccionado del catálogo Siemens S.A. Su función dentro del simulador es permitir al usuario elegir entre uno de los 2 telemotores para el manejo del navío.

Tabla 2.7*Característica de un selector 3 posiciones*

Figura	Características
	<p>N° de referencia: 3SU1130-2BL60-1NA0</p> <p>Marca: Siemens</p> <p>Voltaje (Regulable): 5-500 [V_{DC}]</p> <p>Corriente: 10 [A]</p>

Nota. Datos obtenidos de la ficha técnica del componente [22].

La Tabla 2.15 muestra el selector de 3 posiciones del catálogo de Siemens S.A. Dentro del diseño de la consola de mando del simulador, se utilizan dos selectores de 3 posiciones. A continuación, se detalla la función de cada uno:

- Selector 1: Permite al usuario elegir entre dos bombas para el funcionamiento del navío. La posición central permite elegir usar ambas bombas a la vez.
- Selector 2: Permite al usuario elegir entre los modos de manejo. El primero es el manejo manual, que se realiza mediante el timón y la palanca de mando, el segundo es el modo de manejo con joystick (NFU), y, finalmente, la última posición permite elegir un modo de manejo automático.

Tabla 2.8*Característica de la palanca de mando NORISYS 4 LS4*

Figura	Características
	<p>Marca: Noris Automation</p> <p>Voltaje (Regulable): 18-32 [V_{DC}]</p> <p>Corriente: 0.5-1.5 [A]</p> <p>Señal de salida analógica estándar: 4-20 [mA]</p>

Nota. Datos obtenidos de la ficha técnica [23].

La Tabla 2.16 palanca de mando de Noris es permitir el control de las turbinas del navío mediante dos palancas, una para cada turbina. Este componente se activa en el modo de manejo manual.

Tabla 2.9*Características del joystick de la consola de mando*

Figura	Características
	Marca: SENTOP Modelo: JP50 Voltaje máximo de entrada: 48 [V _{DC}] Corriente máximo de entrada: 5 [A]

Nota. Datos obtenidos de la ficha técnica [24].

La función del joystick de mando SENTOP es permitir el control la dirección del navío. Este componente se activa cuando el modo de manejo seleccionado es NFU.

Tabla 2.10*Volante con potenciómetro*

Figura	Características
	N° de referencia: A40122000377 Resistencia interna: 5 [KΩ] Marca: Industrias Lorenzo, S.A. Potencia: 2 [W]

Nota. Datos obtenidos de la ficha técnica [25].

La tabla 2.10 muestra el volante con potenciómetro de Industrias Lorenzo. S.A. La función del volante es permitir el control de la dirección del navío como el timón. Este componente se activa cuando el modo de manejo seleccionado es manual. Se utiliza como un control redundante, debido a que, en las misiones es necesario contar con diferentes alternativas de funcionamiento en caso de un fallo imprevisto.

Tabla 2.11*Televisor TCL 50S450G*

Figura	Características
	Marca: TCL Potencia de consumo: 75 [W] Dimensiones: 50 [in] Resolución: 4K

Nota. Datos obtenidos de la ficha técnica [26].

En la tabla 2.11 se muestran los televisores cuya función es representar las ventanas para que el usuario visualice el entorno virtual desarrollado. La elección se basó en el tamaño estimado de las ventanas del centro de comando del navío y en función de la calidad de imagen deseada.

Tabla 2.12*Monitor para consola de mando AOC 20E1H HD*

Figura	Características
	Marca: AOC Potencia de consumo: 50 [W] Sistema de operación: Windows y Linux Dimensiones: 20 [in] Resolución: HD

Nota. Datos obtenidos de la ficha técnica [27].

La tabla 2.12 muestra el monitor seleccionado para la visualización de la consola de mando. En esta consola se muestra la ventana principal del simulador donde el usuario podrá elegir las condiciones de la práctica y posteriormente empezar la simulación.

Tabla 2.13*Características recomendadas de CPU*

Figura	Características
	<p>Procesador: AMD Ryzen 7</p> <p>Disco duro: 2 [TB] HDD</p> <p>Disco sólido: 1 [TB] SSD</p> <p>Memoria RAM: 16 [GB]</p> <p>Tarjeta gráfica: NVIDIA GE-FORCE RTX 3060</p> <p>Potencia: 450 [W]</p>

Nota. Datos obtenidos de la ficha técnica [28].

En la Tabla 2.13, se muestra el CPU elegido para ejecutar el entorno virtual. Se buscó que el CPU cumpla con las características para simular el entorno de manera eficiente y fluida, para conseguir una experiencia lo más cercana a la real.

2.5.3 Selección de fuente de poder

Para la selección de la fuente de poder, se tomó en cuenta, el voltaje necesario para el funcionamiento de todos los actuadores, así como del controlador. La Figura 2.6 muestra la fuente seleccionada de 24 [V] a 2 [A].

Figura 2.6

Fuente de poder seleccionada



Nota. Fuente seleccionada para alimentar los componentes [29]

2.5.4 Diagrama eléctrico

En la Figura 2.7 se muestra las conexiones realizadas en el proyecto. En el diagrama de conexiones que se presenta, los componentes son adaptados mediante arreglos eléctricos para conectarlos con el microcontrolador seleccionado.

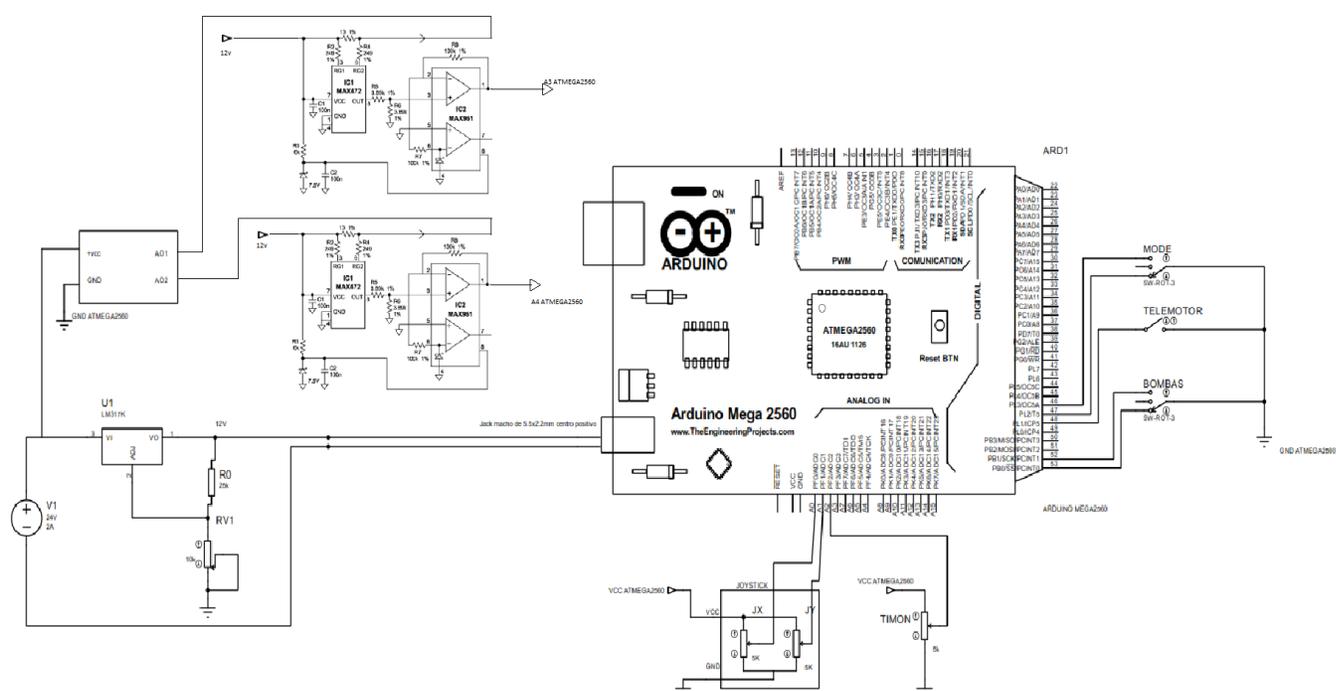
Estos arreglos eléctricos no son necesarios para el manejo de los selectores de 3 y 2 posiciones, debido a que, se los utiliza solamente como *switches* entre la señal digital del pin seleccionado y los 0 [V].

En el caso del timón y el joystick, internamente funcionan con un potenciómetro de 5[k Ω] que puede conectarse directamente al microcontrolador.

Para el controlador diferencial, si bien la salida es dada como una señal analógica, es necesario adaptarlo debido a la corriente que consume supera a la que puede recibir el ATMEGA2560; por ello, se optó por alimentarlo mediante una fuente externa.

Figura 2.7

Diagrama de conexiones del proyecto



Nota. El diagrama esquemático realizó en el *software* de Proteus.

2.5.5 Adaptación de señales

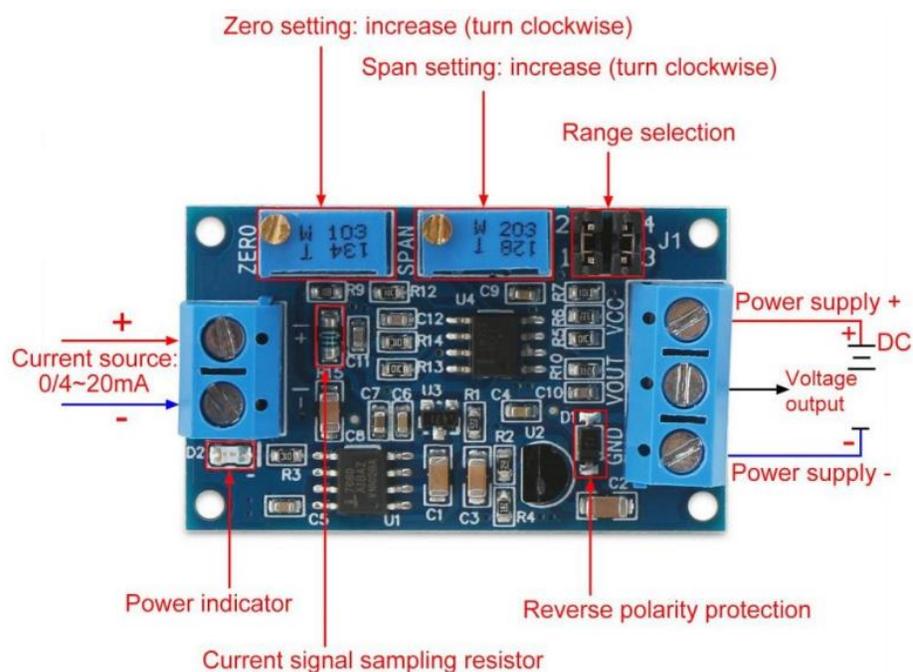
Para el procesamiento de la señal enviada por la palanca de control diferencial, se necesita un convertidor de corriente a voltaje para que el Arduino pueda manejarlo. Este convertidor permite recibir las señales de 4 a 20 [mA] enviados por el controlador diferencial y escalarla dentro de un

rango de 0 a 5 [V]. Con el valor escalado, es posible enviar esta nueva salida a los pines analógicos del Arduino, para manejar el comportamiento del modelo creado en el entorno virtual.

La Figura 2.8 muestra este módulo, que es utilizado ampliamente para lograr conectar sensores de grado industrial con microcontroladores que trabajan en función de voltajes más pequeños.

Figura 2.8

Módulo convertidor de corriente a voltaje



Nota. Módulo convertidor para el escalamiento de amperaje [30].

2.6 Parámetros de diseño mecánico

En esta sección se muestra el diseño de la embarcación, el diseño de la consola de mandos y la incorporación de los actuadores. Además, se presenta un plano de implantación. que permita ubicar la consola en la posición óptima dentro del puente de gobierno de la Fragata.

2.6.1 Diseño de embarcación

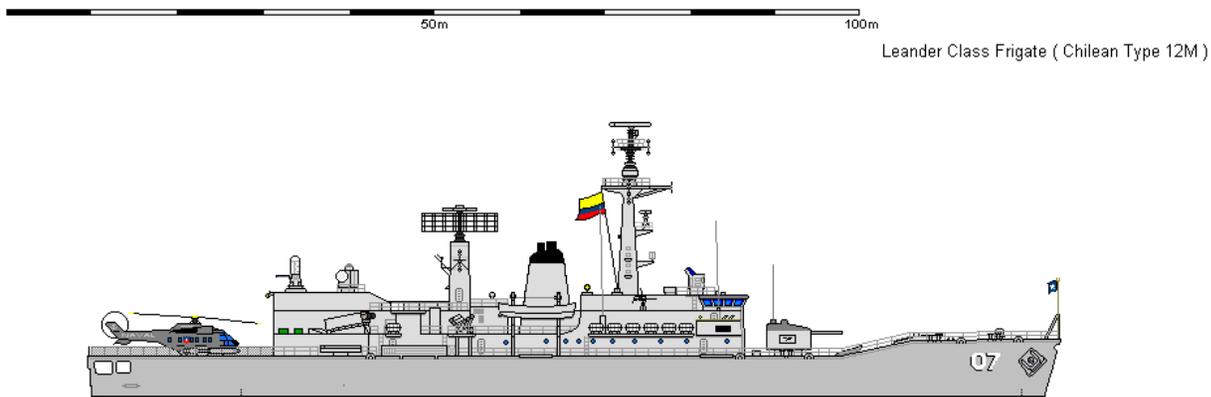
El tipo de embarcación que se consideró en esta simulación fue el buque insignia clase Leander de la Armada del Ecuador del escuadrón de fragatas misileras llamado FM-01 Presidente Eloy Alfaro.

En 1973, se construyó este buque en los astilleros Yarrow (Shipbuilders) en Scoutsf Glasgow, Escocia. En ese mismo año, este buque proporcionó servicio a la Armada de Chile bajo el nombre de Almirante Condell.

En el año de 1993, ASMAR (Astilleros de la Armada de Chile) modifico la armería y sus instalaciones. A fines del año 2007 e inicios del 2008, este buque fue transferido a la Armada del Ecuador bajo el nombre de FM-01 Presidente Eloy Alfaro. La Figura 2.9 muestra un dibujo a escala del navío.

Figura 2.9

Dibujo a escala de una fragata tipo Leander



Nota. Diseño obtenido de libros de la Armada del Ecuador [31].

En las Tabla 2.14, Tabla 2.15 y Tabla 2.16, se puede observar información detallada sobre las características generales, dimensionamiento y peso, y las capacidades que posee este buque fragata de tipo Leander:

Tabla 2.14*Características generales del buque fragata*

Tipo	PFG Leander Y-160
Desplazamiento a toda carga	3,176.9 [tons]
Autonomía	4,500 [MN] a 12 [nudos]
Velocidad máxima (220 RPM)	28 [nudos]
Velocidad económica	14 [nudos]
Tipo de máquinas motrices	2 turbinas a vapor (2 calderas)
Potencia	30,000 [SHP]
Capacidad de tripulantes	260
Combustible	Diésel

Nota: Datos obtenidos mediante entrevista [32].

El modelo tridimensional de la fragata se realizó en base a los planos obtenidos en libros militares marinos. Además, cabe recalcar que, si bien no encontramos información explícita de la embarcación, el modelo es un diseño a detalle tomando en cuenta la información disponible para acceso público. En la Figura 2.10 se muestra el modelo réplica de la fragata [33].

Tabla 2.15*Dimensiones y peso del buque fragata*

Eslora máxima	113.4 [m]
Manga	14.3 [m]
Puntal	8.49 [m]
Calado a proa	4.1 [m]
Calado a popa	4.4 [m]
Calado con domo	5.5 [m]
Desplazamiento a toda carga	3,186.9 [tons]
Desplazamiento liviano	2,336.8 [tons]

Nota: Datos obtenidos mediante entrevista [32].

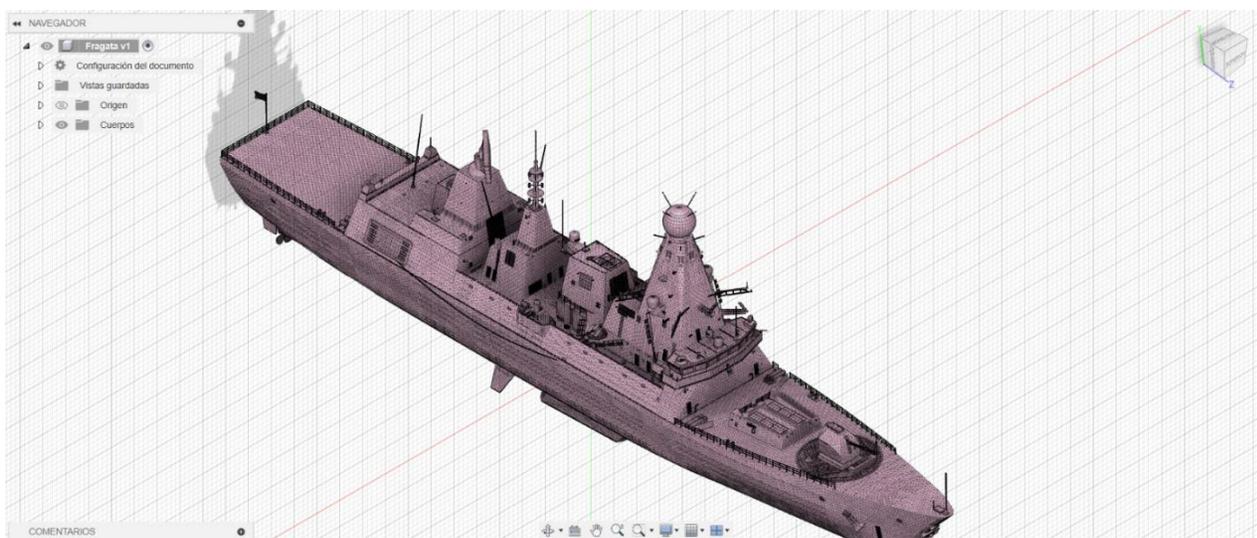
Tabla 2.16*Capacidades del buque fragata*

Diésel (95%)	164,262.9 [gal]
JP1 (95%)	14,956.5 [gal]
Agua de reserva (95%)	10,320.8 [gal]
Agua de bebida (95%)	30,749.6 [gal]
Carga refrigerada	30.0 [m ³]
Generadores principales (turbo)	2
Potencia	500 [kW] c/u

Nota: Datos obtenidos mediante entrevista [32].

Figura 2.10

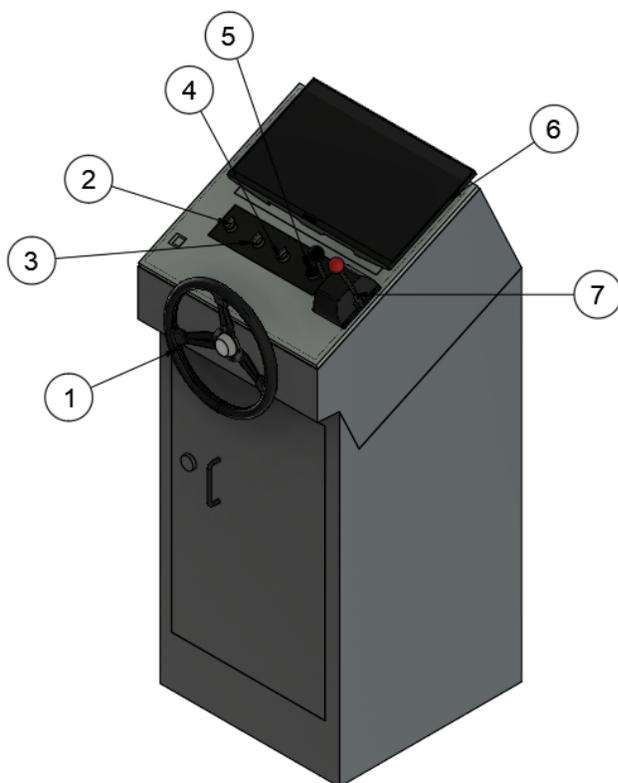
Modelo réplica de la embarcación FRAGATA



Nota: El buque fue diseñado en el software Fusión 360.

2.6.2 Diseño de la consola de mando

La Figura 2.11 consola de mando para este proyecto se ha diseñado a partir de los requerimientos solicitados por el cliente. Esta consola de mando se ensambló con los diferentes equipos mencionados en el apartado de selección de componentes eléctricos. Cabe recalcar que las medidas de esta consola de mando fueron obtenidas de las diferentes hojas técnicas de los componentes; a más de ello, ciertos componentes se obtuvieron en forma de archivos con extensión step de las propias páginas de los fabricantes.

Figura 2.11*Diseño de la consola de mando*

LISTA DE PIEZAS		
ELEMENTO	NOMBRE DE PIEZA	DESCRIPCIÓN
1	Timón de barco	Modo 1: Manejo de barco en entorno virtual
2	Selector dos posiciones	Selección de telemotores
3	Selector tres posiciones	Selección entre las dos bombas de barco
4	Selector tres posiciones	Selección entre modos de manejo
5	Joystick de control	Modo 2: Manejo de barco en entorno virtual
6	Monitor principal	Presentación de ventana principal del simulador
7	Palanca de mando	Control de turbinas

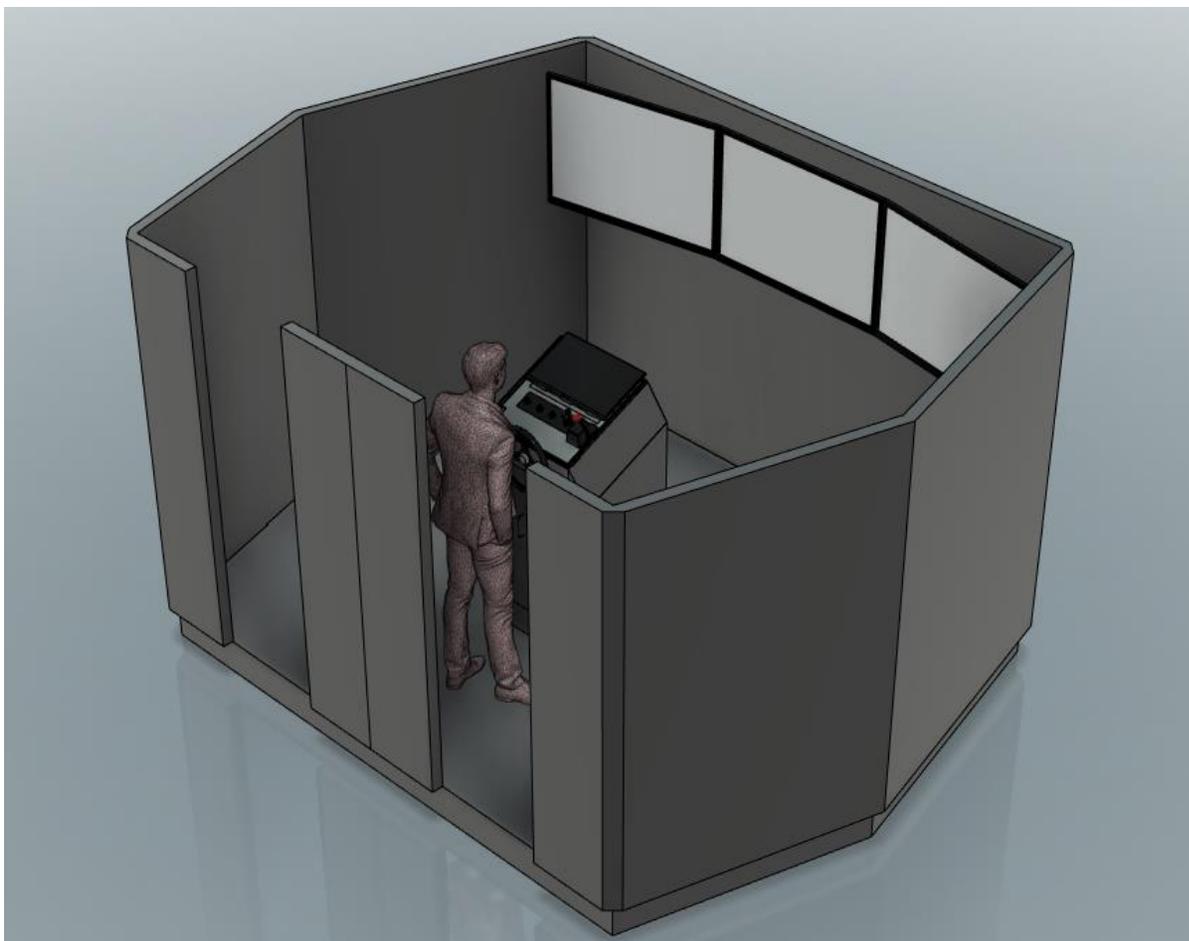
Nota: La consola de mando diseñada en Fusion 360

2.6.3 Plano de implantación

La Figura 2.12 muestra el plano de implantación se presenta la ubicación que la consola y los monitores tendrían que ser colocados en el puente de gobierno del buque militar fragata; ya que, el lugar donde estén ubicados cada uno de los tres monitores y la distancia que tiene entre estos monitores con la consola es de importancia.

Figura 2.12

Diseño de distribución del simulador de navegación y la consola.



Nota: El diseño de la sección del puente de mando se realizó en Fusion 360.

En el plano de implantación se tomó en consideración medidas referenciales de una entrevista realizada al capitán de navío en estado inactivo Imaicela [32], conocedor de buques marinos militares, que al entrevistarlo proporcionó información relevante acerca de las medidas del buque fragata misilera FM-01 Presidente Eloy Alfaro. El plano de implantación se encuentra en el apéndice F. En el plano se muestra que la medida referencial entre los televisores y la consola de

mando es de 1.5[m], esta distancia es prudente para que el practicante a timonero pueda visualizar el entorno virtual en las pantallas y experimente de manera correcta la simulación.

2.7 Parámetros de diseño de programación

En este apartado se presenta la selección de Entorno de Desarrollo Integrado (IDE) para el diseño del entorno virtual y la plataforma que recibe las señales de los componentes externos para la comunicación entre ellos en Unity.

2.7.1 Selección de Entorno de Desarrollo Integrado (IDE)

En la selección de IDE se consideró las limitaciones y beneficios de los entornos de programación. Se consideró dos tipos de IDE: Microsoft Visual Studio y Visual Studio Code; por el cual, se analizó las características de ambas plataformas.

Tabla 2.17*Diferencias entre plataformas de programación*

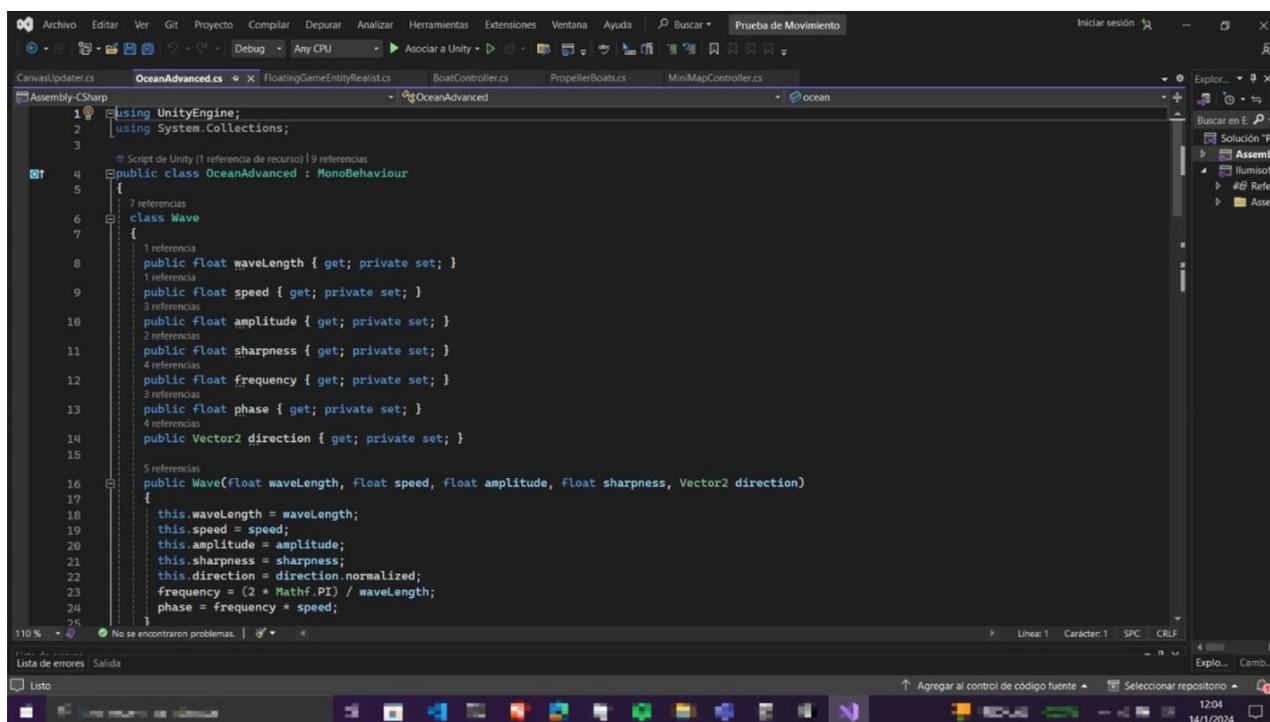
Categoría	Microsoft Studio Visual	Visual Studio Code
Tipo	IDE robusta para escribir, editar, depurar y ejecutar códigos.	Editor de texto de código abierto para escribir y editar códigos.
Ejecutable	Windows y Mac.	Windows, Mac y Linux.
Tamaño de instalación	Se descargar más de 40 GB en Windows y 6 GB en Mac.	No requiere más de 200 MB en cualquier plataforma
Soporte	Integra C# y .NET. compatible con otros lenguajes de programación como: Python, C++, lenguajes web y más.	Integra JavaScript, TypeScript y Node JS. Para otros lenguajes de programación se usan extensiones.
Precios	Es gratis la categoría comunidad. La categoría profesional y empresarial poseen un costo de \$45 y \$250 USD/mes individualmente.	Es gratis, al igual que sus extensiones. Hay extensiones <i>freemium</i> .
Extensiones	No incorpora muchas extensiones	Posee muchas extensiones profesionales que se adquieren considerando los requerimientos del programador.

Nota. Los datos fueron obtenidos de la página web *freecodecamp* [34].

En la Tabla 2.17 se detallan las características de cada IDE, en base a la información, Microsoft Visual Studio surgió como la mejor opción para el código de programación utilizado en Unity, debido a su flexibilidad para los desarrolladores. A parte, Microsoft Visual Studio cuenta con diversos tipos de categorías, en el proyecto se utilizó la comunitaria debido a que es gratuita. La Figura 2.13 muestra el entorno del IDE seleccionado.

Figura 2.13

Plataforma de Microsoft Visual Studio enlazado a Unity



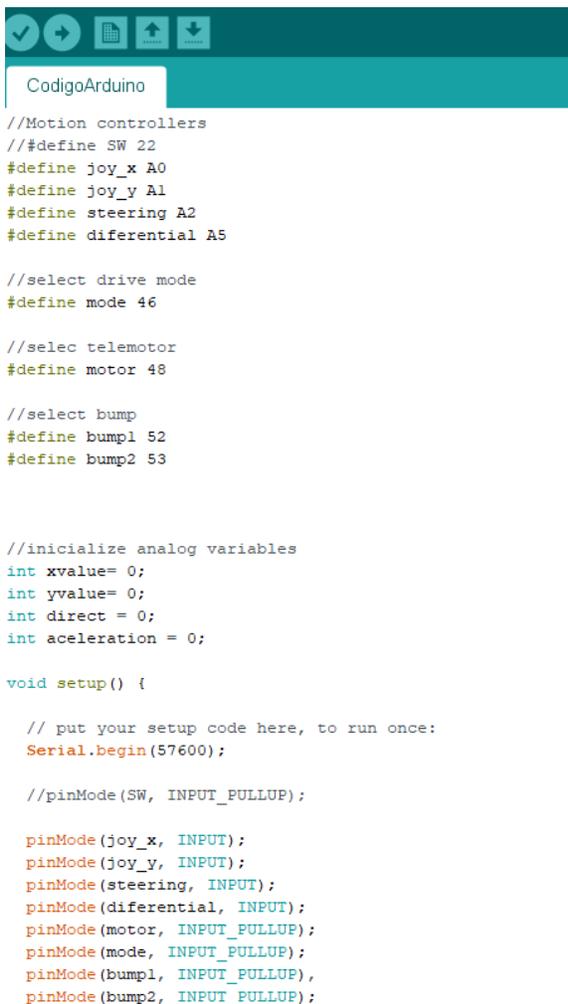
Nota: En la figura se muestra parte del código de programación utilizado en Microsoft Visual Studio para el desarrollo del entorno virtual.

En lo que respecta al procesamiento de señales para los componentes electrónicos, se utilizó el IDE perteneciente a Arduino. Recordando el tipo de placa electrónica con el que se trabajó este

proyecto, que fue el Arduino MEGA, los más viable y factible fue utilizar su propio IDE de código abierto, éste se muestra en la Figura 2.14.

Figura 2.14

Software de Arduino



```

CodigoArduino
//Motion controllers
//#define SW 22
#define joy_x A0
#define joy_y A1
#define steering A2
#define diferencial A5

//select drive mode
#define mode 46

//selec telemotor
#define motor 48

//select bump
#define bump1 52
#define bump2 53

//inicialize analog variables
int xvalue= 0;
int yvalue= 0;
int direct = 0;
int aceleration = 0;

void setup() {

  // put your setup code here, to run once:
  Serial.begin(57600);

  //pinMode(SW, INPUT_PULLUP);

  pinMode(joy_x, INPUT);
  pinMode(joy_y, INPUT);
  pinMode(steering, INPUT);
  pinMode(diferencial, INPUT);
  pinMode(motor, INPUT_PULLUP);
  pinMode(mode, INPUT_PULLUP);
  pinMode(bump1, INPUT_PULLUP);
  pinMode(bump2, INPUT_PULLUP);
}

```

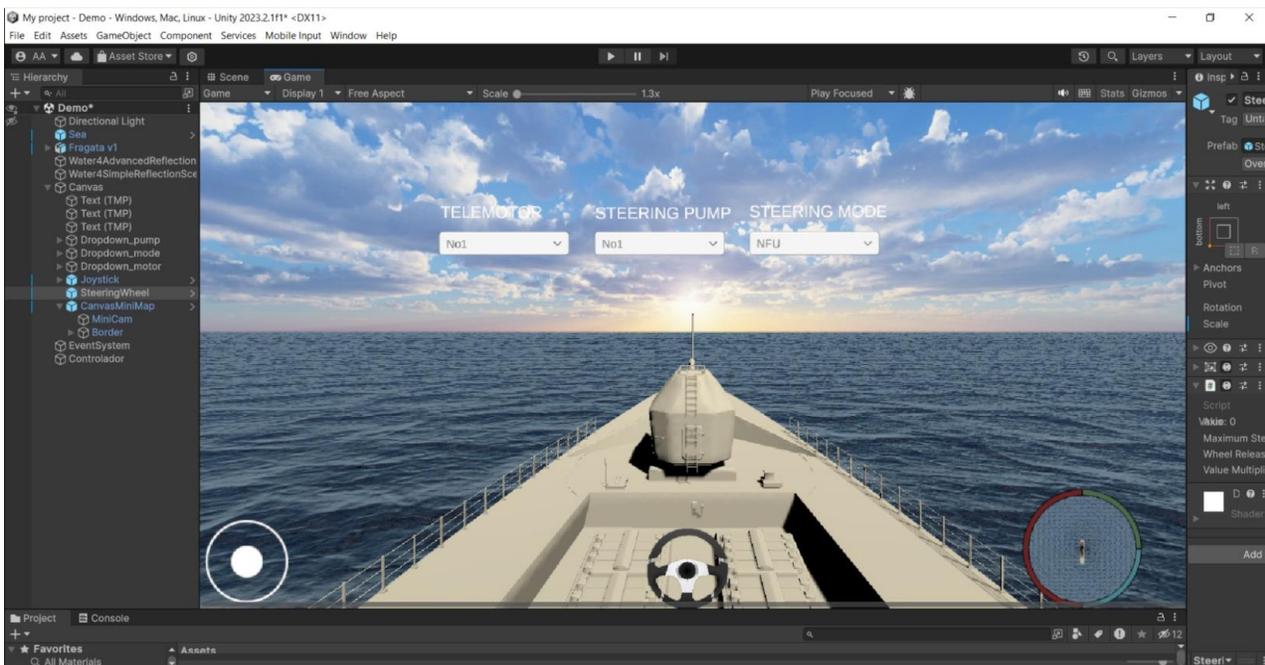
Nota. En la figura se muestra parte del código de programación utilizado en el IDE de Arduino para la recepción y transmisión de datos de los componentes físico al entorno virtual.

2.7.2 Diseño del entorno virtual

Para el diseño del entorno virtual en Unity, se empezó por la creación del océano y se realizó el Script o código de programación que permita emular de la forma más parecida el comportamiento del océano real.

Figura 2.15

Pantalla principal de Unity

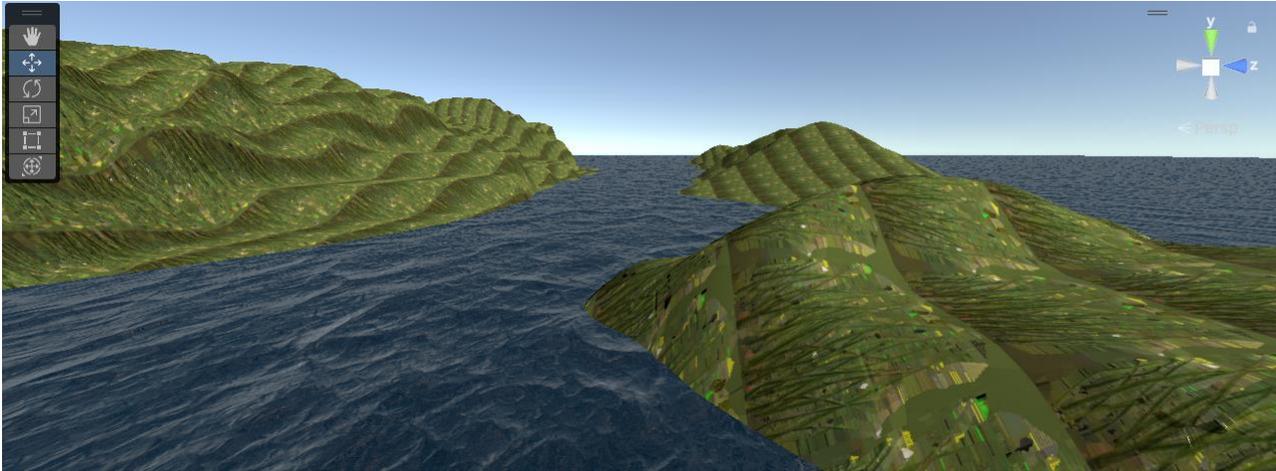


Nota. Pantalla principal del simulador de navegación realizado en Unity.

Una vez realizado el océano y el comportamiento que posee, se importó el modelo 3D de la embarcación (Figura 2.15). Con el modelo de un ambiente marino dentro del entorno virtual, se diseñó una parte del entorno virtual que emule el buscado por el cliente, siendo este un brazo de salida al mar (Figura 2.16).

Figura 2.16

Imagen del ambiente virtual de Unity



Nota. La figura muestra un brazo de salida al mar para que el barco diseñado navegue y se mueva por medio de los actuadores.

2.7.3 Algoritmos de navegación y condiciones de manejo.

Para que el modelo del barco implementado en el entorno virtual ejecute el movimiento deseado al igual que las distintas condiciones de manejo para los selectores, se realizó la codificación de las físicas de movimiento mediante Scripts programados en lenguaje C#. Se crearon diferentes Scripts utilizando el paradigma orientado a objetos, para una mejor separación de los diferentes métodos a implementar.

Para ello, se creó un Script que permita realizar el movimiento del bote mediante el uso de fuerzas aplicadas sobre el objeto. Este código permite un movimiento traslacional, así como rotacional del navío en el océano, al mismo tiempo que, interactúa con el comportamiento de las olas.

Para la traslación, se aplicaron fuerzas de empuje en función de un valor de velocidad y aceleración descrito dentro del código. Además, dicha velocidad depende del número de propulsores que posea el modelo del navío.

El movimiento rotacional del navío se realizó mediante un proceso similar, sin embargo, en este caso la rotación se calcula como la interpolación de la posición actual del navío con la posición marcada por el potenciómetro o el joystick mediante el uso de cuaterniones, si bien este método no es el más sencillo, pero si es el que permite representar giros en un espacio tridimensional de manera más realista.

Capítulo 3

3 Resultados y análisis

En este capítulo se presentan los resultados y análisis de la solución propuesta, abordando los objetivos planteados al inicio del proyecto. Para ello, se toma en cuenta tres aspectos fundamentales: diseño de la consola de mando, diseño del entorno virtual, y la interacción del entorno con elementos físicos. Este último punto se valida mediante una implementación emulando el funcionamiento del simulador con actuadores de bajo costo que representen los seleccionados en el capítulo 2.

Finalmente, se presentan el rubro con los costos necesarios para la implementación del proyecto y se incorpora un análisis evaluando el impacto del proyecto junto con una aproximación de la inversión necesaria para su desarrollo.

3.1 Diseño de consola de mando

Como se indicó en el capítulo 1, uno de los objetivos fue diseñar la consola de comando a utilizar en el simulador. Para el diseño de la consola de mando, se realizó en base a la visita que se tuvo con el cliente. En las primeras etapas del proyecto, se tomaron medidas referenciales de una consola de mando real que el cliente poseía y, de aquella consola de mando presentada, se determinaron los actuadores necesarios a emplear.

En el apéndice D de este proyecto, se muestran las diferentes medidas establecidas para la consola de mando del simulador de navegación.

3.2 Diseño del entorno virtual

El diseño del entorno virtual corresponde a la creación de un ambiente que permita manejar el navío a través del mar. Además, se diseñó y agregó una superficie terrestre que permita demostrar la interacción del navío, no solo con el océano sino con otras superficies.

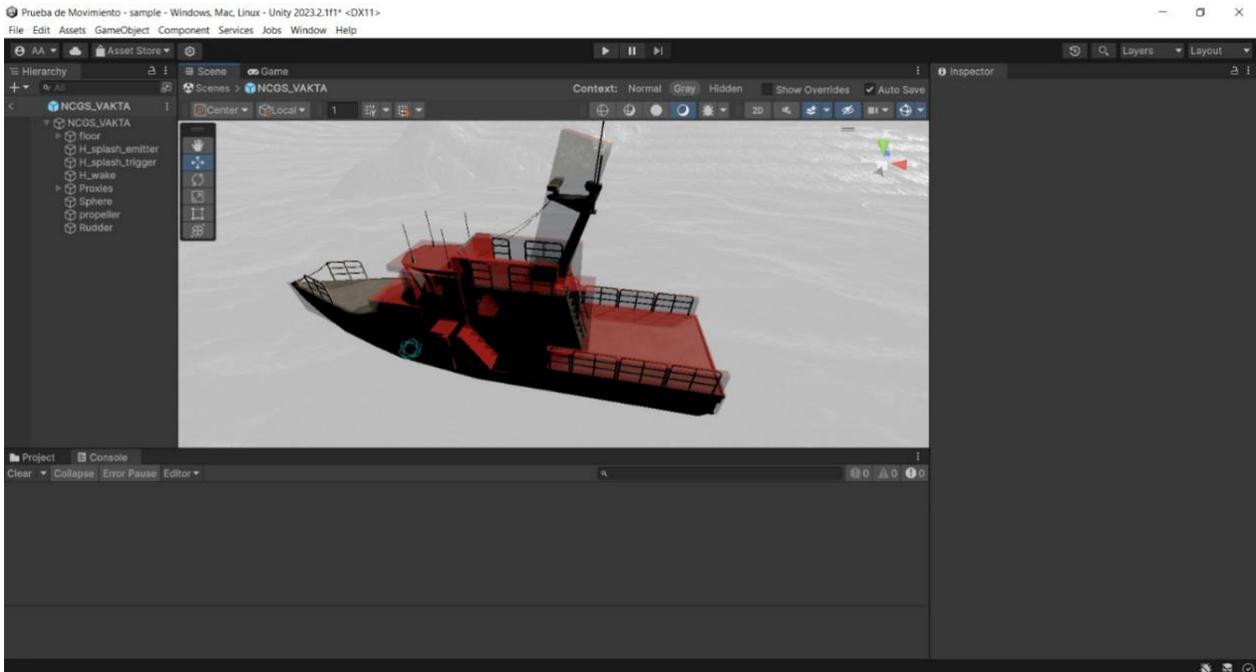
El principal uso que el cliente requiere del simulador es la posibilidad de crear diferentes entornos en base a mapas oceánicos reales. Por ello, se desarrolló un entorno que emule un brazo de mar y permita navegar el navío a través de él.

3.3 Interacción del entorno con actuadores físicos.

El funcionamiento del simulador se validó mediante el uso de actuadores de bajo costo disponibles en el mercado ecuatoriano, tomando en cuenta el tiempo de entrega del proyecto y el presupuesto del que dispone. Así mismo, la Figura 3.1 muestra que se utilizó un modelo 3D de una embarcación más simple debido a las limitaciones computacionales de los equipos utilizados para el desarrollo del proyecto [35]; a parte que, al utilizar el modelo de la fragata diseñada representaba un uso de recursos muy altos para el tipo de equipos con los que se desarrollaron este proyecto.

A pesar de ello, el código utilizado en el proyecto se adapta a diversos modelos de embarcaciones que se realicen. La ventaja de realizar la programación de los *scripts* con el paradigma orientado a objetos es que las clases y métodos son escalables y se encuentran segmentados en diferentes archivos de código que hacen más sencilla su interpretación.

En el apéndice B se muestra un video con el entorno y el funcionamiento del control del barco mediante actuadores físicos.

Figura 3.1*Embarcación de baja resolución*

Nota. Embarcación con menor cantidad de nodos que permite la ejecución de movimiento sin exceder los recursos del ordenador utilizado en la simulación.

En el Apéndice A, se puede observar los componentes utilizados para el prototipo de baja resolución. Se usaron componentes de fácil acceso para interactuar con el entorno virtual creado. Es importante mencionar que estos componentes poseen características similares a los componentes industriales; a más de ello, que se los encuentran en cualquier tienda electrónica del Ecuador y posee un bajo costo.

Se presenta también en el Apéndice B un diagrama de conexiones que muestran de manera sencilla las conexiones realizadas en el prototipo de baja resolución, y un video del funcionamiento de la consola con su respectiva respuesta en el entorno virtual.

3.4 Consumo energético

Los datos de corriente, voltaje y potencia se obtuvieron por medio de las fichas técnicas de cada uno de los componentes electrónicos que se seleccionaron para este proyecto se presentan en la Tabla 3.1.

Tabla 3.1

Detalles del consumo energético

Cantidad	Componente	Corriente [A]	Voltaje [V]	Potencia [W]	Horas de uso [h]	Energía [Wh]
1	Arduino Mega 2560	0.5	12	6	4	24
1	Palanca de mando	0.5	24	12	4	48
1	Joystick eléctrico	0.08	5	0.4	4	1.6
1	Timón/volante	0.04	5	0.2	4	8
3	Monitor de presentación	-	-	75	4	300
1	Monitor para consola	-	-	50	4	200
1	CPU	-	-	450	4	1800
1	Convertidor de corriente	0.16	12	1.92	4	7.68
1	Fuente de alimentación	2	24	48	4	192
CONSUMO ENERGÉTICO TOTAL						2589.78

Nota: Se consideró todos los componentes del apartado de componentes de grado industrial seleccionados para el cliente

El consumo energético total del simulador que se tendrá en un día, utilizándolo 4 horas será de 2580.68 [Wh] que es igual a 2.58 [kWh]. Este valor es muy razonable, tomando en cuenta que el mayor consumo proviene del CPU siendo este un componente primordial en el funcionamiento del simulador.

Tabla 3.2

Detalles de costo de componentes

Cantidad	Componente	Valor unitario (\$)	Valor total (\$)
1	Selector 2 posiciones	11.16	11.16
2	Selector 3 posiciones	23.77	47.54
1	Palanca de mando Norisys 4 LS4	787.65	787.65
1	Monitor táctil de 15 pulgadas	359.99	359.99
1	Joystick industrial eléctrico	63.00	63.00
1	Volante	131.12	131.12
3	Televisor	331.00	993.00
1	CPU	1250.00	1250.00
1	GPU NVIDIA GEFORCE RTX-3060	1500.00	1500.00
1	Fuente de poder	20.40	20.40
1	Módulo convertidor	14.70	14.70
1	ATMEGA2560	30.00	30.00
Valor total			5208.56

Nota: Costos de los componentes seleccionados para el simulador.

Es importante decir que la tarjeta gráfica NVIDIA GeForce RTX 3060 es capaz de soportar hasta tres pantallas simultáneamente, permitiendo la visualización del entorno virtual en las pantallas ubicadas en la cabina. Además, su precio puede variar considerablemente según su demanda, actualmente, su uso para el minado de monedas virtuales y entrenamiento de modelos de inteligencia artificial han hecho que aumenten drásticamente su precio.

3.5 Costos

3.5.1 Costos de equipos

En lo que respecta a la sección de costos, se consideraron los valores indicados en la Tabla 3.2, tomando en consideración cada uno de los componentes a utilizar en el proyecto.

3.5.2 Costos de implementación

Para la implementación del simulador se debe tomar en cuenta no solo el costo de los equipos como se muestra en la Tabla 3.3; si no que, también se debe tomar en cuenta que es necesario un equipo de trabajo multidisciplinario que desarrolle el proyecto.

Es necesario contar con al menos dos programadores con bases sólidas en la programación de entornos como el de Unity, estos elementos serán los encargados de desarrollar el entorno virtual y su funcionamiento, tomando en cuenta todos los tipos de movimiento e interacciones requeridas de la manera más realista posible.

Así mismo, es necesario un ingeniero en telecomunicaciones o redes. Para lograr tener el alcance esperado del simulador, es necesario la interconexión de dos o más computadores mediante

una subred ethernet local. Con esto, se consigue que existan dos o más usuarios que puedan interactuar con la simulación al mismo tiempo. En el simulador que se pretendería desarrollar serian el practicante y el evaluador.

De igual manera, es necesario un ingeniero mecatrónico que se encargue del diseño eléctrico y de la adquisición de señales de los componentes físicos y el envío los datos hasta el CPU. Así mismo, se encargará de crear los modelos 3D para adaptarse a otro navío en caso de ser necesario. Finalmente, con el conocimiento del ingeniero mecatrónico en diversas áreas como mecánica, electricidad y programación puede manejar al equipo y mejorar el desempeño al contar con un elemento que pueda ayudar en diferentes rubros.

Tabla 3.3

Detalles de costos de ingeniería

Profesional	Cantidad	Sueldo individual (USD)	Sueldo mensual (USD)
Programador	2	1200 – 1700	2400 – 3400
Redes o sistemas	1	700 – 1000	700 – 1000
Mecatrónico	1	1100 – 1500	1100 – 1500
Total	4	3000 – 4200	4200 – 5900

En la Tabla 3.3 se muestra una estimación de los salarios para las profesiones requeridas en el equipo de desarrollo del simulador de navegación. Los datos de los salarios son estimaciones promedio obtenidas a través de la plataforma Glassdor [36] [37] [38].

Es importante recalcar que, el desarrollo de un proyecto de esta magnitud puede llevar un lapso de alrededor de 1 año. Esto es debido al tipo de sistema que es un sistema de grado militar que requiere tener la precisión y el realismo necesario para que los próximos timoneros puedan prepararse con las mejores condiciones. No obstante, el tiempo de trabajo de cada profesional variará dependiendo del avance del proyecto.

Para el ingeniero en redes o sistemas, se estima un tiempo estimado de 2-3 meses para la interconexión entre los equipos. El ingeniero mecatrónico y los programadores un tiempo de trabajo de entre 7-12 meses debido a que se encargaran de la programación, diseño y simulación del entorno, además El ingeniero mecatrónico se encargara de la interconexión de los actuadores con el entorno virtual. Finalmente, luego de aproximadamente un año de trabajo se debería contar con el ingeniero mecatrónico para el mantenimiento del simulador, ya que es el que posee el conocimiento interdisciplinario para encargarse de los diferentes problemas que puedan surgir.

Capítulo 4

4 Conclusiones y recomendaciones

El proyecto que se presentó tuvo como propósito ser una alternativa económica y autónoma de los simuladores de navegación, debido a que actualmente el sistema depende de entidades externas para su adquisición y mantenimiento.

En la siguiente sección se muestra las diversas conclusiones que se obtuvieron una vez terminado el proyecto y cumpliendo los objetivos propuestos en el capítulo 1; también se mencionan recomendaciones a considerar, en el caso que se desee replicar este proyecto.

4.1 Conclusiones

- La placa electrónica ATMEGA2560 permite manejar actuadores de tipo industrial, siempre y cuando se, realice un estudio meticuloso de las fichas técnicas de los equipos a utilizar y se los conecte correctamente. Se validó este funcionamiento mediante actuadores de bajo coste conectados a la placa electrónica que permitan manejar el navío en el entorno virtual.
- El desarrollo del entorno virtual marítimo en el que su objeto barco es controlado mediante actuadores físicos, permitió la navegación de un navío y el uso de diferentes condiciones de manejo
- El desarrollo de un simulador de navegación de alta resolución propio supone un costo elevado a corto plazo de alrededor de 50000 dólares, tomando un tiempo de trabajo estimado de un año, sin embargo, observándolo desde una perspectiva a largo plazo resulta beneficios en el ámbito de seguridad, económico y tecnológico.

- El uso de fusión 360 permitió crear modelos 3D tanto de los navíos como de la consola de mando. Ambos indispensables para la integración del entorno de simulación con actuadores físicos.
- Contar con un simulador propio es de gran importancia para un ente militar debido a la sensibilidad de la información que se pueda manejar. El uso de un software externo puede disminuir la seguridad en el manejo de información, ya que se depende de entidades externas que podrían tener acceso a los datos.
- La integración de diferentes ingenierías en el proyecto y el gran avance presentado, indica la importancia de los ingenieros mecatrónicos dentro de la industria tecnológica; puesto que, implican un ahorro de tiempo y dinero.
- El entorno virtual de navegación se presenta como una solución óptima para el entrenamiento del personal militar. Mediante actuadores que mantienen la fidelidad de movimiento de embarcaciones marinas, se evitan los posibles perjuicios a la integridad estructural y humana que puedan presentarse en una práctica con un navío real.

4.2 Recomendaciones

- Considerar otros equipos alternativos para la adquisición de datos entre los componentes eléctricos y el entorno virtual, tales como: PLCs (Controlador Lógico Programable) o DAQ (tarjeta de adquisición de datos). El uso de estos componentes puede evitar usar arreglos eléctricos adicionales en el momento de la adquisición de señales.
- Agregar a la simulación un sistema de geolocalización al entorno virtual; puesto que, estos sistemas resultan ser importante al momento de navegar por las diversas estructuras que se encuentran en brazos de mar, altamar y al momento de arribar a puerto.

- Conseguir un equipo multidisciplinario que ejecute los siguientes pasos en el proceso de desarrollo, como son: continuar con la programación del entorno virtual, programación de la interfaz de usuario para el evaluador, la conexión de los actuadores de nivel industrial con el entorno y el desarrollo de la subred local que interconecta el entorno del practicante con el del evaluador.

Referencias

- [1] Armada de Chile, «Capacidades,» CIMAR. Centro de Instrucción y Capacitación Marítima, [En línea]. Available: <https://www.cimar.cl/cimar/simulacion/capacidades>.
- [2] D. Rodríguez , «Simuladores: realismo a favor de la seguridad vial y la eficiencia,» Revista Transportes y Turismo, 19 Septiembre 2018. [En línea]. Available: <https://www.tyt.com.mx/nota/simuladores-realismo-a-favor-de-la-seguridad-vial-y-la-eficiencia>.
- [3] E. R. González Guimerá, «Simuladores Náuticos y sus usos,» 2018. [En línea]. Available: <https://riull.ull.es/xmlui/handle/915/14571>.
- [4] Virtual Marine Technology (VMT), «Simulador de navegación MissionQuest - Fast Rescue Craft Training,» Nautic Expo, [En línea]. Available: <https://www.nauticexpo.es/prod/virtual-marine-technology-inc/product-33876-310546.html>.
- [5] H. Enrique, «Armada del Ecuador,» Foro Militar General, 1 Mayo 2020. [En línea]. Available: <https://www.militar.org.ua/foro/viewtopic.php?p=7486069>.
- [6] S. Straus, M. Lewis, K. Connor, R. Eden , M. Boyer, T. Marler, C. Carson, G. Grimm y H. Smigowski, Colletive Simulation-Based Training in the U.S. Army, Santa Monica: Rand Corporation, 2019.
- [7] J. Stergiou , «Modeling and simulation for navy platform design,» de *Conferencia de American Society of Naval Engineers (ASME)*, 2022.

- [8] U.S. Navy, «Youtube,» 09 Enero 2020. [En línea]. Available: <https://www.youtube.com/watch?v=meo59q03x0w>. [Último acceso: 20 noviembre 2023].
- [9] LEONARDO DRS, «Conjuntos de terminales de enlace de datos tácticos (TDL),» [En línea]. Available: <https://www.leonardodrs.com/what-we-do/products-and-services/tactical-data-link-tdl-terminal-sets/>.
- [10] M. Everly, «El personal de las instalaciones del simulador de vuelo proporciona una formación realista a los pilotos,» 28 Agosto 2015. [En línea]. Available: https://www.army.mil/article/154521/Flight_simulator_facility_staff_provides_realistic_training_for_pilots/.
- [11] GDIT, «Simulador marítimo VirtualShip,» [En línea]. Available: <https://www.gdit.com/perspectives/our-stories/maritime-simulation-virtualship/>.
- [12] M. Velichko, «VR Military Training – the Next Step of Combat Evolution,» Jasonren Look on the bright side, [En línea]. Available: <https://jasoren.com/vr-military-training-the-next-step-of-combat-evolution/>.
- [13] VSTEP, «Simulador naval NAUTIS,» 7 Diciembre 2018. [En línea]. Available: <https://www.vstepsimulation.com/nautis-simulator/nautis-maritime-simulator/naval/>.
- [14] ASNE, «AMERICAN SOCIETY OF NAVAL ENGINEERS,» 08 septiembre 2022. [En línea]. Available: <https://www.navalengineers.org/Education/Modeling-Sim2022>. [Último acceso: 21 noviembre 2023].

- [15] L. J. P. Buitiriago, «UNAB,» 2021. [En línea]. Available: https://repository.unab.edu.co/bitstream/handle/20.500.12749/14215/2021_Tesis_Lesly_Perlaza_Buitrago.pdf?sequence=1. [Último acceso: 21 noviembre 2023].
- [16] H. J. H. P. Yunjoo Kim, «IEEE,» 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Honolulu, HI, USA, 2018.
- [17] I. Burn, «Arduino: qué es, para qué sirve y cómo funciona,» 19 07 2022. [En línea]. Available: <https://es.ccm.net/ordenadores/hardware/4488-arduino-que-es-para-que-sirve-y-como-funciona/>.
- [18] Dynamo electronics, «Board programable Ref MEGA2560 R3 marca Arduino Original,» [En línea]. Available: <https://www.dynamoelectronics.com/tienda/arduino-mega-2560-r3/>.
- [19] M. Bayer, «El entrenamiento y la simulación tocan todos los aspectos de las fuerzas armadas de EE. UU.,» National Defense, 30 Octubre 2023. [En línea]. Available: <https://www.nationaldefensemagazine.org/articles/2023/10/30/training-simulation-touches-every-aspect-of-us-military>.
- [20] ARDUINO, «Mega 2560 Rev3,» [En línea]. Available: <https://docs.arduino.cc/hardware/mega-2560>.
- [21] Tienda Electrica Virtual, «Selector de dos posiciones 1NA 3SU1130-2BF60-1BA0,» [En línea]. Available: <https://www.tiendaelectricavirtual.com/producto/mando-y-senalizacion-3su-3su1130-2bf60-1ba0/>.

- [22] Siemens S.A, «Siemens,» [En línea]. Available: <https://mall.industry.siemens.com/mall/en/en/Catalog/product?mlfb=3SU1130-2BL60-1NA0>.
- [23] NORIS, «CONTROL LEVERS: Control lever for propulsion drives NORISYS 4 LT4,» NORIS, [En línea]. Available: <https://www.noris-group.com/products-solutions/automation-components/propulsion-control-components/control-levers/norisys-4-lt4/>.
- [24] Sentop, «Direct Industry,» [En línea]. Available: <https://www.directindustry.es/prod/sentop-by-shanghai-sibo-m-e-co-ltd/product-235548-2389679.html>. [Último acceso: enero 2024].
- [25] Industrias Lorenzo, S.A.U., «VOLANTE 270 POTENCIÓMETRO Y ARO,» [En línea]. Available: <https://www.industrias-lorenzo.com/es/volantes/118-volante-270-potenciometro.html>.
- [26] DisplaySpecifications, «DisplaySpecifications,» [En línea]. Available: <https://www.displayspecifications.com/en/model/73973384>.
- [27] AOC, «AOC,» [En línea]. Available: <https://aoc.com/us/products/monitors/20e1h>. [Último acceso: enero 2024].
- [28] Mercado Libre, «Cpu Gamer Ryzen 7 /ram 16gb/hdd 2tb+1tb Ssd/ Nvidia Rtx 8gb,» [En línea]. Available: https://articulo.mercadolibre.com.ec/MEC-518415304-cpu-gamer-ryzen-7-ram-16gbhdd-2tb1tb-ssd-nvidia-rtx-8gb-_JM#is_advertising=true&position=1&search_layout=stack&type=pad&tracking_id=9ccea_e99-e49f-48f3-a083-393643765b3f&is_advertising=true&ad_domain=VQCAT.

- [29] TETTSA, «TETTSA,» [En línea]. Available: <https://tienda.tettsa.gt/producto/fuente-de-alimentacion-conmutada-24vdc-2a/>. [Último acceso: enero 2024].
- [30] Tinkersphere, «Current to Voltage Converter Module,» Tinkersphere, [En línea]. Available: <https://tinkersphere.com/sensors/3319-current-to-voltage-converter-module.html>.
- [31] Armada del Ecuador, «Características principales,» de *Guía General de la Unidad Conoce tu Buque FRAMOR*.
- [32] V. Imaicela Rosillo, Interviewee, *Información acerca del buque fragata Eloy Alfaro*. [Entrevista]. 7 Enero 2024.
- [33] MagicCGIStudios, 22 octubre 2022. [En línea]. Available: <https://www.turbosquid.com/es/3d-models/hms-daring-royal-navy-type-45-class-air-defence-destroyer-model-1805684>. [Último acceso: 15 octubre 2023].
- [34] freeCodeCamp, «Visual Studio vs Visual Studio Code: ¿Cuál es la diferencia entre estos editores de código IDE?,» 3 Marzo 2023. [En línea]. Available: <https://www.freecodecamp.org/espanol/news/visual-studio-vs-visual-studio-code-cual-es-la-diferencia-entre-estos-editores-de-codigo-ide/>.
- [35] N. Gauer, «Unity Asset Store,» 31 mayo 2017. [En línea]. Available: <https://assetstore.unity.com/packages/tools/physics/fast-buoyancy-61079>. [Último acceso: 21 novimebre 2023].

- [36] Glassdor, «Glassdor,» [En línea]. Available: https://www.glassdoor.com/Salaries/quito-ingeniero-mecatr%C3%B3nico-salary-SRCH_IL.0,5_IC2217055_KO6,27.htm. [Último acceso: 15 1 2023].
- [37] Glassdor, «Glassdor,» [En línea]. Available: https://www.glassdoor.com.mx/Sueldos/quito-software-developer-sueldo-SRCH_IL.0,5_IM1362_KO6,24.htm. [Último acceso: 15 1 2023].
- [38] Glassdor, «Glassdor,» [En línea]. Available: https://www.glassdoor.com.ar/Sueldos/ecuador-ingeniero-de-telecomunicaciones-sueldo-SRCH_IL.0,7_IN68_KO8,39.htm. [Último acceso: 15 1 2023].
- [39] SAISAC MECATRONICA, «Cable jumper 40cm/40 pines macho-hembra,» [En línea]. Available: <https://saisac.pe/producto/cable-jumper-40cm-40-pines-macho-hembra/>.
- [40] UDVABONY, «KY023 JOYSTICK Module Dual Axis XY Joysticks 5 Pin Leads 5V Joysticks Module PS2 Joystick Control Lever Sensor Game Controller,» [En línea]. Available: <https://udvabony.com/product/joystick-module-for-arduino/>.
- [41] TESLAELECTRONIC, «Potenciómetro 10k Ohm,» [En línea]. Available: <https://www.teslaelectronic.com.pe/producto/potenciometro-10k-ohm/>.
- [42] TEKNOMOVO, «Switch de palanca 3 posiciones on/off/on,» [En línea]. Available: <https://teknomovo.com.mx/producto/switch-de-palanca-3-posiciones-on-off-on/>.

- [43] Revista Científica General José María Córdova, «La realidad virtual como herramienta para la educación básica y profesional,» 1 Enero 2021. [En línea]. Available: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1900-65862021000100223.
- [44] M. Vidal Ledo , R. Avello Martínez, M. Rodríguez Monteagudo y J. A. Menéndez Bravo, «Simuladores como medios de enseñanza,» 2019. [En línea]. Available: <https://www.medigraphic.com/pdfs/educacion/cem-2019/cem194j.pdf>.
- [45] GDIT, «Ready for whatever tomorrow brings.,» [En línea]. Available: <https://www.gdit.com/capabilities/mission-services/training-and-simulation/>.
- [46] J. Melús , «El enorme potencial de los simuladores de conducción,» ITCL Centro Tecnológico, 17 Enero 2023. [En línea]. Available: <https://itcl.es/blog/el-enorme-potencial-de-los-simuladores-de-conduccion/>.
- [47] J. Córdova, «Herramienta virtual para simular maniobras con busques tipo GNL en la bahía de Cartagena,» Scielo, 2018. [En línea]. Available: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1900-65862021000100223.
- [48] W. Rodríguez , «Tecnologías marítimas: Simuladores,» 21 11 2018. [En línea]. Available: https://www.researchgate.net/publication/371451124_Tecnologias_maritimas_Simuladores/link/648468e679a722376524acf8/download.
- [49] J. D. González Torrejón y I. A. Pérez Vivian, «Importancia del simulador de navegación y maniobra en la escuela naval del Perú,» 2014. [En línea]. Available: <https://repositorio.escuelanaval.edu.pe/bitstream/handle/20.500.14101/109/Importancia%20>

de1%20Simulador%20de%20Navegaci%C3%B3n%20y%20Maniobra%20en%20la%20ESNA.pdf?sequence=1.

- [50] Universidad Andrés Bello, «Simulador de Navegación de Marina Mercante,» [En línea]. Available: <https://vinculacion.unab.cl/servicios/simulador-de-navegacion-de-marina-mercante/>.
- [51] Northeast Maritime Institute, «Northeast Maritime Institute Now Offering 100% Online Radar Observer Course,» 18 Marzo 2021. [En línea]. Available: <https://nmi.edu/online-radar-observer-course/>.
- [52] M. M. Sutiono S.Kom., «Arduino #1: Introducción a Arduino,» DonsenIT, [En línea]. Available: <https://dosenit.com/microcontroller/arduino-1-pengenalan-arduino>.
- [53] ARDUINO, «Arduino® MEGA 2560 Rev3 (Datasheet),» ARDUINO, [En línea]. Available: <https://docs.arduino.cc/resources/datasheets/A000067-datasheet.pdf>.
- [54] RS, «Interruptor selector Siemens 3 Posiciones Roscado, 10A, Configuración 2 NA,» [En línea]. Available: <https://es.rs-online.com/web/p/selectores/1355649>.
- [55] B-COMMAND, «Joysticks para hasta 4 posiciones,» B-COMMAND, [En línea]. Available: <https://www.b-command.com/es/productos/joysticks-estacion-de-control-de-joysticks/joystick-mc-mw/>.
- [56] 3nStar, «Monitor táctil capacitivo de 15" sin bisel,» 3nStar, [En línea]. Available: [https://3nstar.com/producto/monitor-de-pantalla-tactil-capacitiva-sin-bisel-15-tcm010vh/?lang=es#:~:text=Sin%20bisel%20Panel%20capacitivo%20sin%20bisel%20de%](https://3nstar.com/producto/monitor-de-pantalla-tactil-capacitiva-sin-bisel-15-tcm010vh/?lang=es#:~:text=Sin%20bisel%20Panel%20capacitivo%20sin%20bisel%20de%20)

2010,t% C3% A1ctil% 20USB% 20Base% 20de% 20monitor% 20de% 20construcci% C3% B3n
% 20met% C3% A1lica.

Apéndices

Apéndice A

Elementos electrónicos seleccionados para prototipo de baja resolución

En este apartado se revisó meticulosamente cada uno de los componentes industriales a implementar en este proyecto; por ello, se optó en buscar componentes económicos, de fácil adquisición, y más que nada que, simulen el actuar de los componentes industriales

Tabla A.1

Elementos seleccionados para la elaboración de prototipo de baja resolución

Componentes	Nombre	Características
	Jumpers	Cables utilizados para las múltiples conexiones entre los módulos electrónicos [39].
	Joystick de Arduino	<p>Dispositivo analógico usado para controlar los movimientos en los ejes X y Y. [40]</p> <p>Voltaje: 3.3 - 5 [V_{DC}]</p> <p>Potenciómetro interno: 10 [KΩ]</p> <p>Temperatura de funcionamiento: 0-70 [°C]</p>

Componentes	Nombre	Características
	Potenciómetro	Componente electrónico de ajuste con un orificio pasante. Resistencia de $10K\Omega$ con un ajuste mayor a un giro [41]. Se lo utiliza dentro del proyecto como timón y control diferencial.
	Switch tipo palanca (selectores dos y tres posiciones)	Switch de tres posiciones on/off/on y dos posiciones on/off. [42].

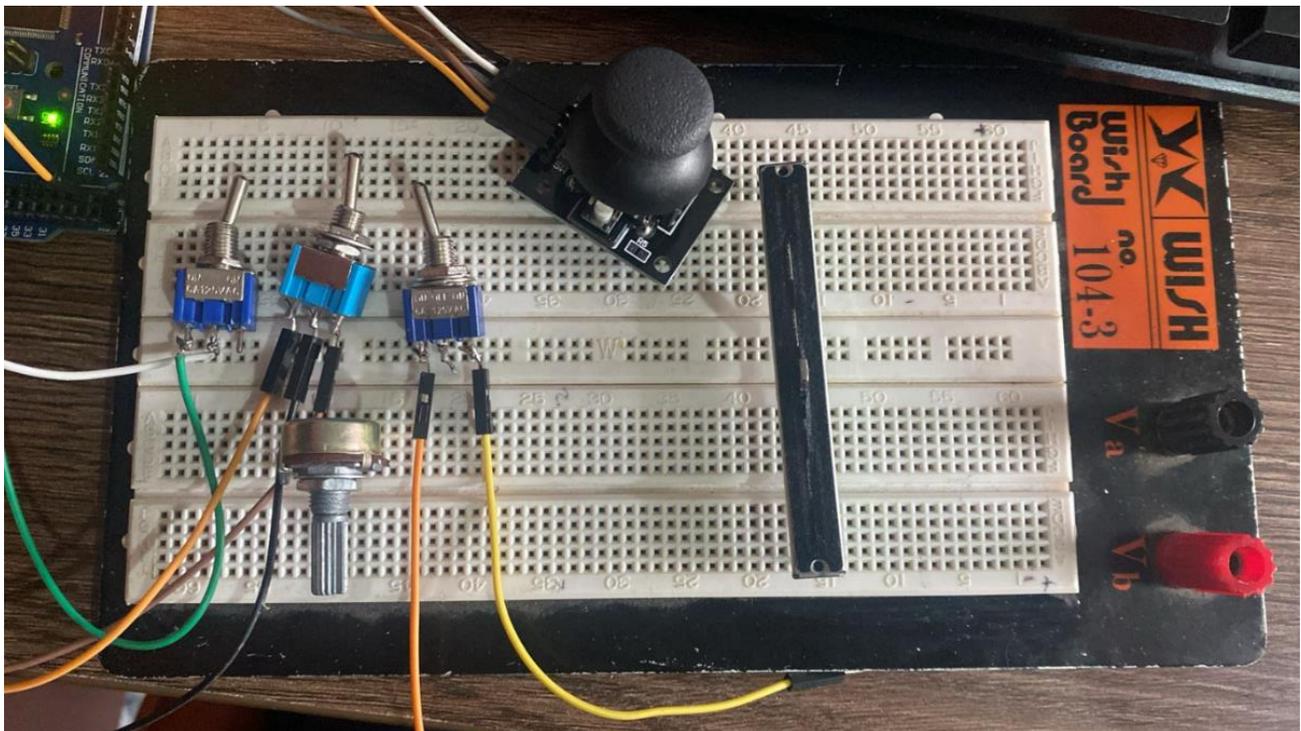
Apéndice B

Prototipo de baja resolución

En esta sección se presenta una imagen que demuestra el prototipo de baja resolución elaborado. Los selectores de dos y tres posiciones, y el joystick fueron soldados a cables jumpers en donde se los conectaron de forma directa a la placa electrónica Arduino Mega; en cambio, el potenciómetro se conectó a un protoboard para luego ser conectado al Arduino Mega.

Figura B.1

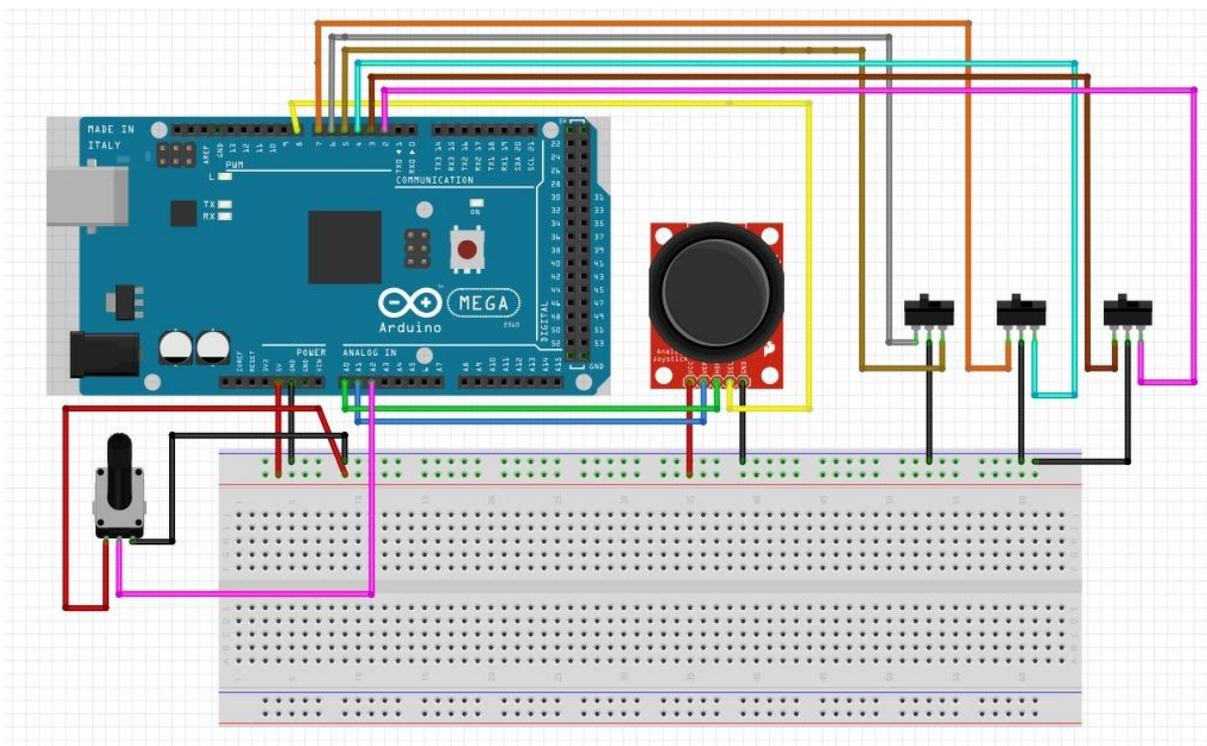
Conexiones físicas del prototipo de baja resolución



En la Figura B.2 se presenta un diagrama esquemático de las conexiones que se realizaron de forma real. En este diagrama se indican los pines utilizados en el Arduino Mega y su correcta conexión con los terminales de los componentes seleccionados en el Apéndice A.

Figura B.2

Diagrama esquemático del prototipo de baja resolución



Video del simulador:

[Simulador.mp4](#)

Apéndice C

Embarcación de baja resolución

En lo que respecta a la embarcación de baja resolución, se optó por utilizar una embarcación que contenga una menor cantidad de nodos; es decir, una embarcación que contenga menos detalles que el buque fragata. La fragata al ser una embarcación con un largo de 113.4 [m] (de popa a proa) y un ancho de 14.3 [m], indican que es una embarcación de gran escala. A más de ello, la cantidad de habitaciones que posee supone un mayor detalle al diseño, obligando al programa de Unity a incrementar los recursos necesarios del ordenador.

Figura C.1

Embarcación utilizada en Unity, mostrando la posición en los diferentes planos X,Y y Z

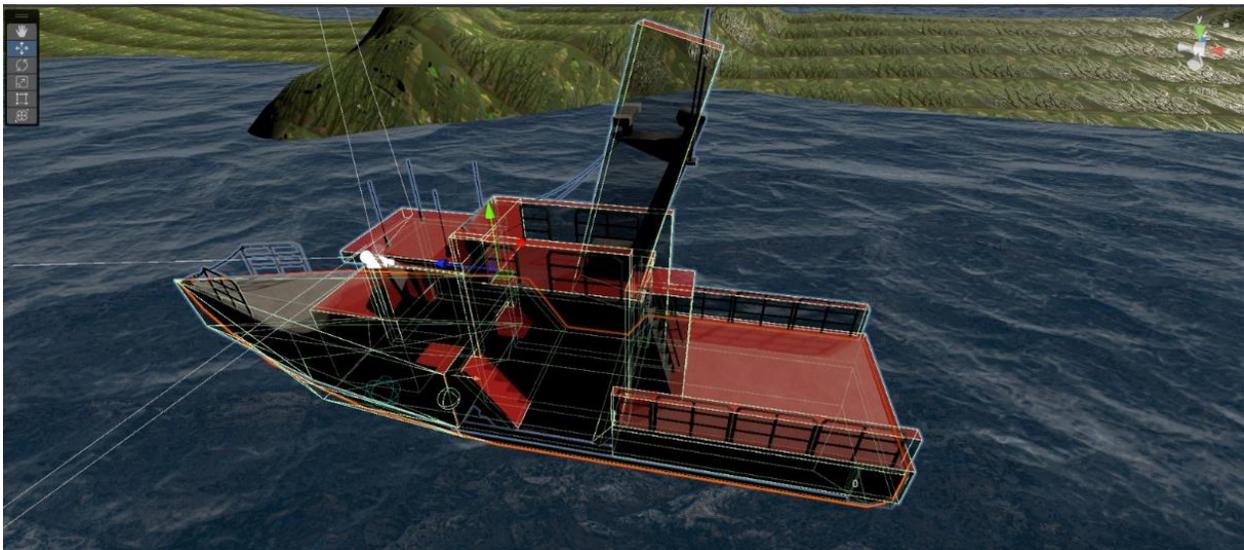
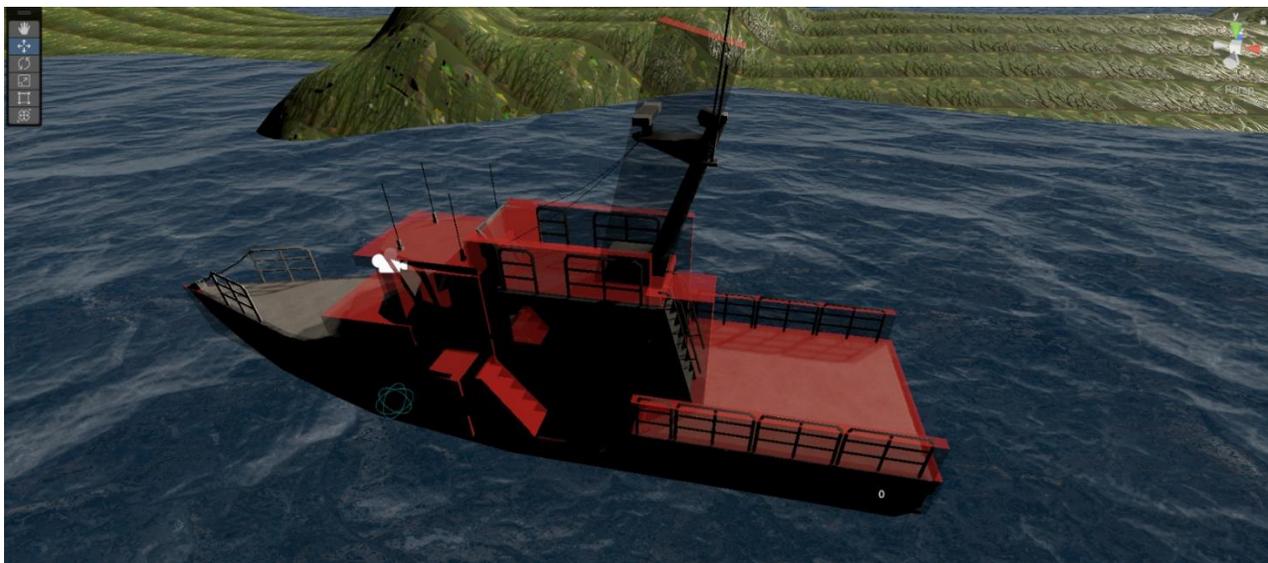


Figura C.2

Embarcación de pesca con menor cantidad de detalles



Apéndice D

Código de programación en el IDE de Microsoft Visual Studio

Figura D.1

Script del océano

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class OceanAdvanced : MonoBehaviour
5 {
6     class Wave
7     {
8         public float waveLength { get; private set; }
9         public float speed { get; private set; }
10        public float amplitude { get; private set; }
11        public float sharpness { get; private set; }
12        public float frequency { get; private set; }
13        public float phase { get; private set; }
14        public Vector2 direction { get; private set; }
15
16        public Wave(float waveLength, float speed, float amplitude, float sharpness, Vector2 direction)
17        {
18            this.waveLength = waveLength;
19            this.speed = speed;
20            this.amplitude = amplitude;
21            this.sharpness = sharpness;
22            this.direction = direction.normalized;
23            frequency = (2 * Mathf.PI) / waveLength;
24            phase = frequency * speed;
25        }
26    };
27
```

```

27
28     public Material ocean;
29     public Light sun;
30
31     private int interaction_id = 0;
32     private Vector4[] interactions = new Vector4[NB_INTERACTIONS];
33
34
35     const int NB_WAVE = 5;
36     const int NB_INTERACTIONS = 64;
37     static Wave[] waves =
38     {
39         new Wave(99, 1.0f, 1.4f, 0.9f, new Vector2(1.0f, 0.2f)),
40         new Wave(60, 1.2f, 0.8f, 0.5f, new Vector2(1.0f, 3.0f)),
41         new Wave(20, 3.5f, 0.4f, 0.8f, new Vector2(2.0f, 4.0f)),
42         new Wave(30, 2.0f, 0.4f, 0.4f, new Vector2(-1.0f, 0.0f)),
43         new Wave(10, 3.0f, 0.05f, 0.9f, new Vector2(-1.0f, 1.2f))
44     };
45
46     void Awake()
47     {
48         Vector4[] v_waves = new Vector4[NB_WAVE];
49         Vector4[] v_waves_dir = new Vector4[NB_WAVE];
50         for (int i = 0; i < NB_WAVE; i++)
51         {
52             v_waves[i] = new Vector4(waves[i].frequency, waves[i].amplitude, waves[i].phase, waves[i].sharpness);
53             v_waves_dir[i] = new Vector4(waves[i].direction.x, waves[i].direction.y, 0, 0);
54         }
55
56         ocean.SetVectorArray("waves_p", v_waves);
57         ocean.SetVectorArray("waves_d", v_waves_dir);
58

```

```

59         for (int i = 0; i < NB_INTERACTIONS; i++)
60             interactions[i].w = 500.0f;
61         ocean.SetVectorArray("interactions", interactions);
62         ocean.SetVector("world_light_dir", -sun.transform.forward);
63     }
64
65     Mensaje de Unity | 0 referencias
66     void FixedUpdate()
67     {
68         ocean.SetVector("world_light_dir", -sun.transform.forward);
69         ocean.SetVector("sun_color", new Vector4(sun.color.r, sun.color.g, sun.color.b, 0.0f));
70     }
71
72     2 referencias
73     public void RegisterInteraction(Vector3 pos, float strength)
74     {
75         interactions[interaction_id].x = pos.x;
76         interactions[interaction_id].y = pos.z;
77         interactions[interaction_id].z = strength;
78         interactions[interaction_id].w = Time.time;
79         ocean.SetVectorArray("interactions", interactions);
80         interaction_id = (interaction_id + 1) % NB_INTERACTIONS;
81     }
82
83     5 referencias
84     static public float GetWaterHeight(Vector3 p)
85     {
86         float height = 0;
87         for (int i = 0; i < NB_WAVE; i++)
88             height += waves[i].amplitude * Mathf.Sin(Vector2.Dot(waves[i].direction, new Vector2(p.x, p.z)) * waves[i].frequency + Time.time * waves[i].phase);
89         return height;
90     }

```

Figura D.1

Script del Navío: flotación

```

1  using UnityEngine;
2  using System.Collections;
3  using System.Collections.Generic;
4  using ArchimedsLab;
5
6
7  [RequireComponent(typeof(MeshFilter))]
8  // Script de Unity (3 referencias de recurso) | 0 referencias
9  public class FloatingGameEntityRealist : GameEntity
10 {
11     public Mesh buoyancyMesh;
12
13     /* These 4 arrays are cache array, preventing some operations to be done each frame. */
14     tri[] _triangles;
15     tri[] worldBuffer;
16     tri[] wetTris;
17     tri[] dryTris;
18     //These two variables will store the number of valid triangles in each cache arrays. They are different from array.Length !
19     uint nbrWet, nbrDry;
20
21     WaterSurface.GetWaterHeight realist = delegate (Vector3 pos)
22     {
23         const float eps = 0.1f;
24         return (OceanAdvanced.GetWaterHeight(pos + new Vector3(-eps, 0F, -eps))
25             + OceanAdvanced.GetWaterHeight(pos + new Vector3(eps, 0F, -eps))
26             + OceanAdvanced.GetWaterHeight(pos + new Vector3(0F, 0F, eps))) / 3F;
27     };

```

```

28 // Mensaje de Unity | 4 referencias
29 protected override void Awake()
30 {
31     base.Awake();
32
33     //By default, this script will take the render mesh to compute forces. You can override it, using a simpler mesh.
34     Mesh m = buoyancyMesh == null ? GetComponent<MeshFilter>().mesh : buoyancyMesh;
35     //Setting up the cache for the game. Here we use variables with a game-long lifetime.
36     WaterCutter.CookCache(m, ref _triangles, ref worldBuffer, ref wetTris, ref dryTris);
37 }
38 // Mensaje de Unity | 4 referencias
39 protected override void FixedUpdate()
40 {
41     base.FixedUpdate();
42     if (rb.IsSleeping())
43         return;
44     /* It's strongly advised to call these in the FixedUpdate function to prevent some weird behaviors */
45
46     //This will prepare static cache, modifying vertices using rotation and position offset.
47     WaterCutter.CookMesh(transform.position, transform.rotation, ref _triangles, ref worldBuffer);
48
49     /*
50     Now mesh ae represented in World position, we can split the mesh, and split tris that are partially submerged.
51     Here I use a very simple water model, already implemented in the DLL.
52     You can give your own. See the example in Examples/CustomWater.
53     */
54     WaterCutter.SplitMesh(worldBuffer, ref wetTris, ref dryTris, out nbrWet, out nbrDry, realist);
55     //This function will compute the forces depending on the triangles generated before.
56     Archimeds.ComputeAllForces(wetTris, dryTris, nbrWet, nbrDry, speed, rb);
57 }

```

```
58 #if UNITY_EDITOR
59 //Some visualizations for this buyoancy script.
60 protected override void OnDrawGizmos()
61 {
62     base.OnDrawGizmos();
63
64     if (!Application.isPlaying)
65         return;
66
67     Gizmos.color = Color.blue;
68     for (uint i = 0; i < nbrWet; i++)
69     {
70         Gizmos.DrawLine(wetTris[i].a, wetTris[i].b);
71         Gizmos.DrawLine(wetTris[i].b, wetTris[i].c);
72         Gizmos.DrawLine(wetTris[i].a, wetTris[i].c);
73     }
74
75     Gizmos.color = Color.yellow;
76     for (uint i = 0; i < nbrDry; i++)
77     {
78         Gizmos.DrawLine(dryTris[i].a, dryTris[i].b);
79         Gizmos.DrawLine(dryTris[i].b, dryTris[i].c);
80         Gizmos.DrawLine(dryTris[i].a, dryTris[i].c);
81     }
82 }
83 #endif
84 }
85
```

Figura D.1

Script del Navío: propulsores

```
1 using UnityEngine;
2 using UnityEditor;
3 using static UnityEditor.Progress;
4 using System;
5
6 Script de Unity (1 referencia de recurso) | 1 referencia
7 public class PropellerBoats : MonoBehaviour
8 {
9     public Transform[] propellers;
10    public Transform[] rudder;
11    private Rigidbody rb;
12
13    9 referencias
14    public float engine_rpm { get; private set; }
15    float throttle;
16    int direction = 1;
17
18    public float propellers_constant = 0.6F;
19    public float engine_max_rpm = 600.0F;
20    public float acceleration_cst = 1.0F;
21    public float drag = 0.01F;
22
23    float angle;
24
25    Mensaje de Unity | 0 referencias
26 void Awake()
27 {
28     engine_rpm = 0F;
29     throttle = 0F;
30     rb = GetComponent<Rigidbody>();
31 }
32 }
```

```

30 void Update()
31 {
32     float frame_rpm = engine_rpm * Time.deltaTime;
33     for (int i = 0; i < propellers.Length; i++)
34     {
35         propellers[i].localRotation = Quaternion.Euler(propellers[i].localRotation.eulerAngles + new Vector3(0, 0, -frame_rpm));
36         rb.AddForceAtPosition(Quaternion.Euler(0, angle, 0) * propellers[i].forward * propellers_constant * engine_rpm, propellers[i].position);
37     }
38
39     throttle *= (1.0F - drag * 0.001F);
40     engine_rpm = throttle * engine_max_rpm * direction;
41
42     angle = Mathf.Lerp(angle, 0.0F, 0.02F);
43     for (int i = 0; i < rudder.Length; i++)
44         rudder[i].localRotation = Quaternion.Euler(0, angle, 0);
45
46     4 referencias
47     public void ThrottleUp()
48     {
49         throttle += acceleration_cst * 0.001F;
50         if (throttle > 1)
51             throttle = 1;
52
53     6 referencias
54     public void ThrottleDown()
55     {
56         throttle -= acceleration_cst * 0.001F;
57         if (throttle < 0)
58             throttle = 0;
59

```

```

60     4 referencias
61     public void Brake()
62     {
63         throttle *= 0.9F;
64     }
65
66     4 referencias
67     public void Reverse()
68     {
69         direction *= -1;
70
71     2 referencias
72     public void RudderRight()
73     {
74         angle -= 0.9F;
75         angle = Mathf.Clamp(angle, -90F, 90F);
76
77     2 referencias
78     public void RudderLeft()
79     {
80         angle += 0.9F;
81         angle = Mathf.Clamp(angle, -90F, 90F);
82
83     Mensaje de Unity | 0 referencias
84     void OnDrawGizmos()
85     {
86         Handles.Label(propellers[0].position, engine_rpm.ToString());
87     }

```

Figura D.1

Script del Navío: controlador de movimiento

```

1  using UnityEngine;
2  using System.Collections;
3  using System;
4  using System.IO.Ports;
5  using UnityEngine.Rendering.VirtualTexturing;
6  using UnityEngine.Windows;
7  using Input = UnityEngine.Input;
8  using static UnityEditor.Progress;
9
10 Script de Unity (1 referencia de recurso) | 1 referencia
11 public class BoatController : MonoBehaviour
12 {
13     public PropellerBoats ship;
14     bool forward = true;
15     SerialPort data = new SerialPort("COM8", 57600);
16     public string receivedString;
17
18     float prevpot = 0.0f;
19
20     // Switch's variables
21     3 referencias
22     public int motor { get; private set; }
23     4 referencias
24     public int mode { get; private set; }
25     4 referencias
26     public int bump1 { get; private set; }
27     4 referencias
28     public int bump2 { get; private set; }

```

```

25 Mensaje de Unity | 0 referencias
26 void Start()
27 {
28     data.Open();
29     data.ReadTimeout = 10;
30 }
31
32 Mensaje de Unity | 0 referencias
33 void Update()
34 {
35     try
36     {
37         if (data.IsOpen)
38         {
39             receivedString = data.ReadLine();
40             Debug.Log(receivedString);
41
42             //Transformar los datos en valores numericos
43             string[] datas = receivedString.Split(',');
44             float movx = float.Parse(datas[0].Trim());
45             float movy = float.Parse(datas[1].Trim());
46             float pot = float.Parse(datas[2].Trim());
47             float aceleration = float.Parse(datas[3].Trim());
48             motor = int.Parse(datas[4].Trim());
49             mode = int.Parse(datas[5].Trim());
50             bump1 = int.Parse(datas[6].Trim());
51             bump2 = int.Parse(datas[7].Trim());
52
53             float difference = pot - prevpot;
54             prevpot = pot;
55
56

```

```

58 // Do the movement
59
60 //mode 0= joystick and mode=1 Steering and diferential
61 if (((bump1 == 1 && bump2 == 0) || (bump1 == 1 && bump2 == 1)) && motor == 1)
62 {
63
64     if (mode == 0)
65     {
66         if (movx < -200) ///| difference< 0 || pot < -450)
67             ship.RudderLeft();
68         if (movx > 200)/// || difference> 0 || pot > 450)
69             ship.RudderRight();
70
71         if (forward)
72         {
73             if (movy > 200)
74                 ship.ThrottleUp();
75             else if (movy < -250)
76             {
77                 ship.ThrottleDown();
78                 ship.Brake();
79             }
80         }
81         else
82         {
83             if (movy < -200)
84                 ship.ThrottleUp();
85             else if (movy > 200)
86             {
87                 ship.ThrottleDown();
88                 ship.Brake();
89             }
90         }

```

```

92         if (!(movy > 200) && !(movy < -200))
93             ship.ThrottleDown();
94
95         if (ship.engine_rpm == 0 && movy < -200 && forward)
96         {
97             forward = false;
98             ship.Reverse();
99         }
100         else if (ship.engine_rpm == 0 && movy > 200 && !forward)
101         {
102             forward = true;
103             ship.Reverse();
104         }
105     }
106
107     else if (mode == 1)
108     {
109         if (difference < 0 || pot < -450)
110             ship.RudderLeft();
111         if (difference > 0 || pot > 450)
112             ship.RudderRight();
113
114         if (forward)
115         {
116             if (aceleration > 50)
117                 ship.ThrottleUp();
118             else if (aceleration < -50)
119             {
120                 ship.ThrottleDown();
121                 ship.Brake();
122             }
123         }
124     }

```

```

125     else
126     {
127         if (aceleration < -50)
128             ship.ThrottleUp();
129         else if (aceleration > 50)
130         {
131             ship.ThrottleDown();
132             ship.Brake();
133         }
134     }
135
136     if (!(aceleration > 50) && !(aceleration < -50))
137         ship.ThrottleDown();
138
139     if (ship.engine_rpm == 0 && aceleration < -50 && forward)
140     {
141         forward = false;
142         ship.Reverse();
143     }
144     else if (ship.engine_rpm == 0 && aceleration > 50 && !forward)
145     {
146         forward = true;
147         ship.Reverse();
148     }
149
150     float variableOriginal = -250.0f; // Ajusta según tu valor original
151     float valorEscalaNueva = EscalarVariable(variableOriginal, -500.0f, 500.0f, 0.0f, 25.0f);
152 }
153 }
154 }
155 else
156 {
157     Debug.Log("no hay un puerto abierto");
158 }

```

```

160     catch (System.Exception e)
161     {
162         e = new System.Exception();
163     }
164 }
165 }
166 }
167
168 1 referencia
169 float EscalarVariable(float valorOriginal, float valorMinimoOriginal, float valorMaximoOriginal, float valorMinimoNuevo, float valorMaximoNuevo)
170 {
171     float valorNormalizado = Mathf.InverseLerp(valorMinimoOriginal, valorMaximoOriginal, valorOriginal);
172     float valorEscalaNueva = Mathf.Lerp(valorMinimoNuevo, valorMaximoNuevo, valorNormalizado);
173     return valorEscalaNueva;
174 }
175 }
176
177 @ Mensaje de Unity | 0 referencias
178 private void OnDestroy()
179 {
180     if (data.IsOpen)
181         data.Close();
182 }
183 }

```

Figura D.1

Script Canvas: Mostrar en pantalla la selección de los selectores

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using TMPro;
4 using UnityEditor.ShaderKeywordFilter;
5 using UnityEngine;
6
7 Script de Unity (1 referencia de recurso) | 0 referencias
8 public class CanvasUpdater : MonoBehaviour
9 {
10     public BoatController controller;
11
12     public TextMeshProUGUI switch1;
13     public TextMeshProUGUI switch2;
14     public TextMeshProUGUI switch3;
15
16     // Start is called before the first frame update
17     Mensaje de Unity | 0 referencias
18     void Start()
19     {
20     }
21
22     // Update is called once per frame
23     Mensaje de Unity | 0 referencias
24     void Update()
25     {
26         if (controller != null)
27         {
28             int motorvalue = controller.motor;
29             int modevalue = controller.mode;
30             int bump1value = controller.bump1;
31             int bump2value = controller.bump2;
```

```
32 switch1.text = "Motor " + (motorvalue+1).ToString() + " seleccionado";
33
34 if (modevalue == 0)
35 {
36     switch2.text = "Modo de manejo con Joystick";
37 }
38 else
39 {
40     switch2.text = "Modo de manejo con timon y acelerador";
41 }
42
43 if (bump1value == 1 && bump2value == 0)
44 {
45     switch3.text = "Bomba 1 seleccionada";
46 }
47 else if (bump1value == 1 && bump1value == 1)
48 {
49     switch3.text = "Ambas bombas seleccionadas";
50 }
51 else
52 {
53     switch3.text = "Bomba 2 seleccionada";
54 }
55
56
57
58
59
60
61
62
```

Apéndice E

Código de programación en el IDE de Arduino

```
//Motion controllers
//#define SW 22
#define joy_x A0
#define joy_y A1
#define steering A2
#define diferencial A3

//select drive mode
#define mode 46

//select telemotor
#define motor 48

//select bump
#define bump1 52
#define bump2 53

//initialize analog variables
int xvalue= 0;
int yvalue= 0;
int direct = 0;
int aceleration = 0;

void setup() {

    // put your setup code here, to run once:
    Serial.begin(57600);

    //pinMode(SW, INPUT_PULLUP);

    pinMode(joy_x, INPUT);
    pinMode(joy_y, INPUT);
    pinMode(steering, INPUT);
    pinMode(diferencial, INPUT);
    pinMode(motor, INPUT_PULLUP);
    pinMode(mode, INPUT_PULLUP);
    pinMode(bump1, INPUT_PULLUP),
    pinMode(bump2, INPUT_PULLUP);
}
```

```
void loop() {

    // put your main code here, to run repeatedly:
    xvalue= analogRead(joy_x);
    yvalue= analogRead(joy_y);
    direct= analogRead(steering);
    aceleration= analogRead(diferential);
    int m = digitalRead(motor);
    int mod= digitalRead(mode);
    int b1= digitalRead(bump1);
    int b2= digitalRead(bump2);

    xvalue= map(xvalue, 0, 1023, -500, 500);
    yvalue= map(yvalue, 0, 1023, -500, 500);
    direct= map(direct, 0, 1023, -500, 500);
    aceleration= map(direct, 0, 1023, -500, 500);

    Serial.print(xvalue);
    Serial.print(",");
    Serial.print(yvalue*-1);
    Serial.print(",");
    Serial.print(direct*-1);
    Serial.print(",");
    Serial.print(aceleration);
    Serial.print(",");
    Serial.print(m);
    Serial.print(",");
    Serial.print(mod);
    Serial.print(",");
    Serial.print(b1);
    Serial.print(",");
    Serial.println(b2);

    //Serial.print(digitalRead(SW));
    //Serial.print(",");

    delay(50);
```

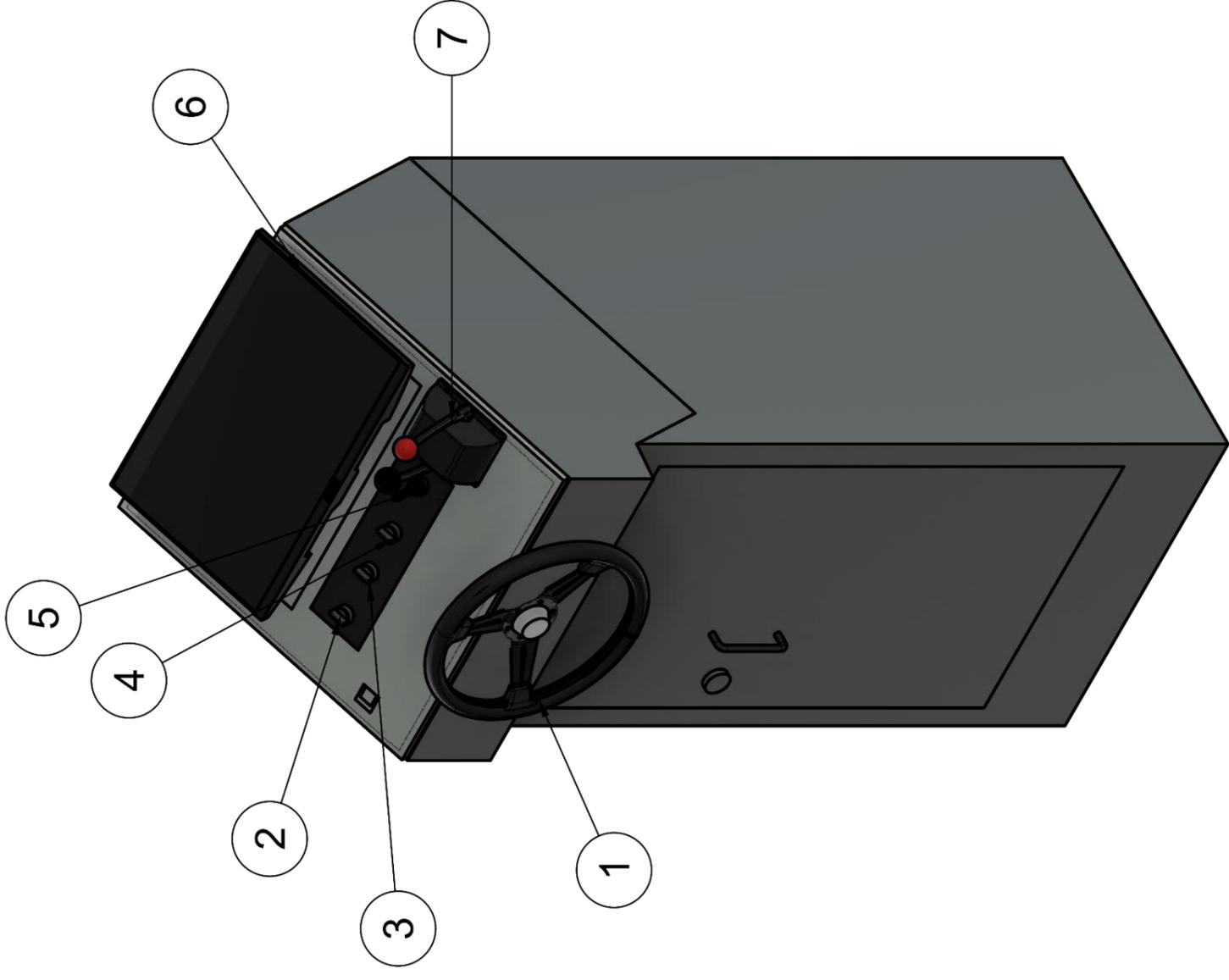
Apéndice F

Planos mecánicos

En esta sección se presentan todos los planos de diseño mecánico que se realizó para este proyecto. Los planos que se realizaron fueron en base a las vistas superior, frontal y lateral en sistema europeo y se tomaron las medidas en cada una de las vistas; cabe nombrar que cada una de las medidas se encuentran en unidades de milímetro. También en los planos se realizó una proyección isométrica del diseño mecánica para apreciar con más detalle las posiciones de las piezas en los ejes X, Y y Z. Es importante nombrar que estos planos son realizador en un tamaño de hoja A3 (297x420 [mm]).

Los planos que se realizaron para este proyecto fueron los siguientes:

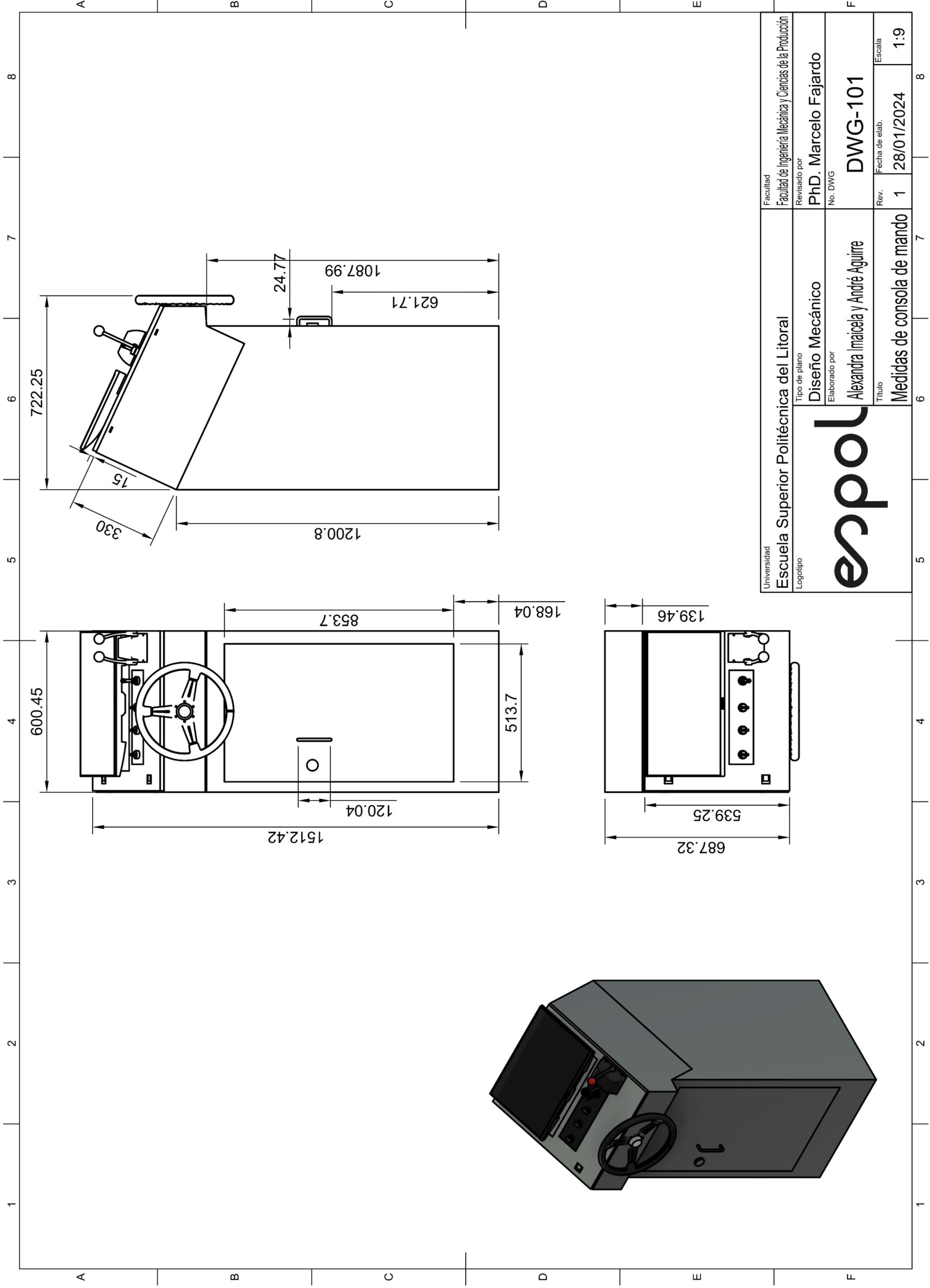
- Lista de piezas
- Medidas de consola de mando
- Plano de implantación



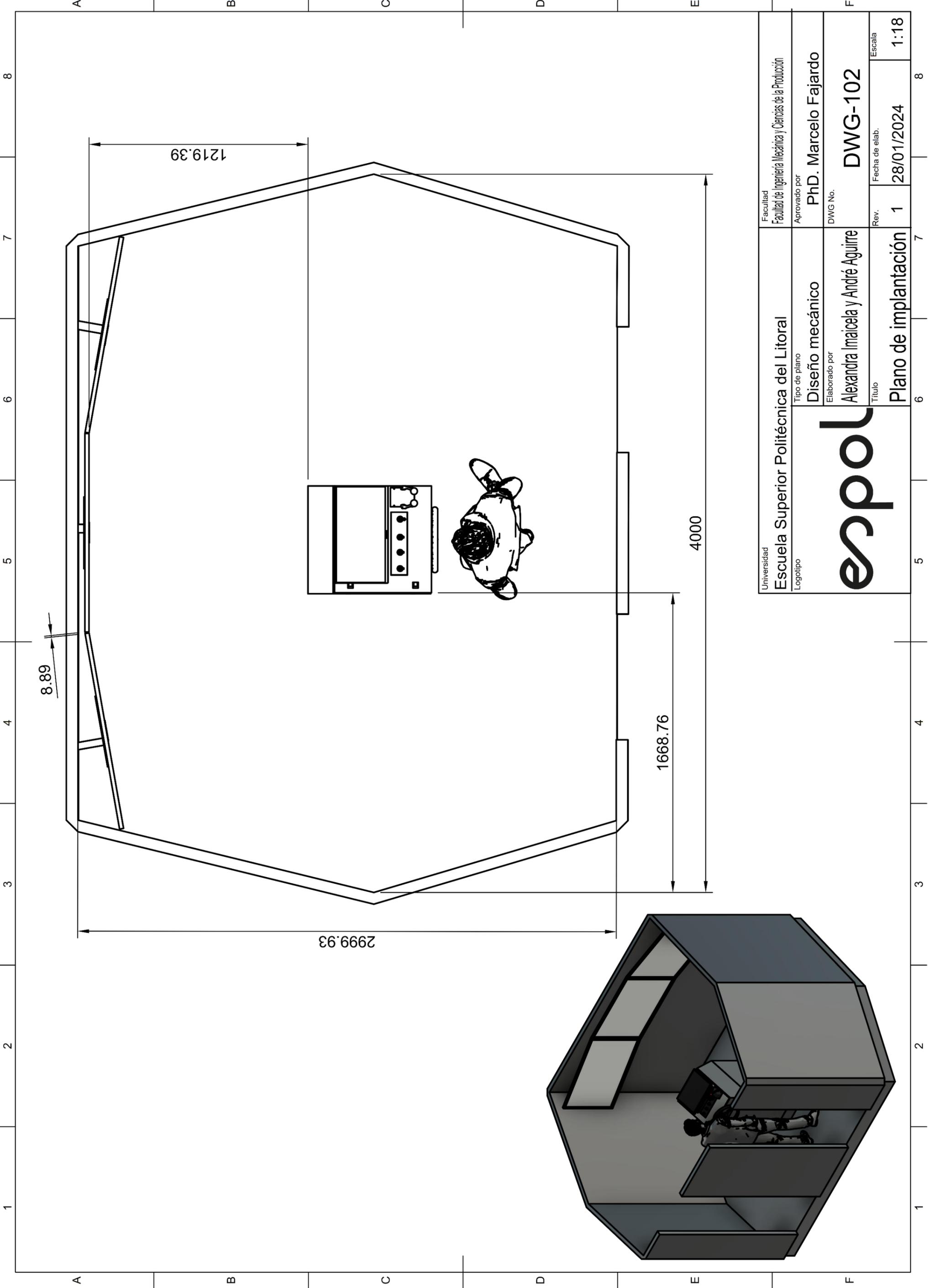
LISTA DE PIEZAS

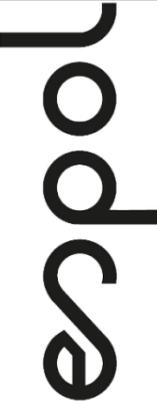
ELEMENTO	NOMBRE DE PIEZA	DESCRIPCIÓN
1	Timón de barco	Modo 1: Manejo de barco en entorno virtual
2	Selector dos posiciones	Selección de telemotores
3	Selector tres posiciones	Selección entre las dos bombas de barco
4	Selector tres posiciones	Selección entre modos de manejo
5	Joystick de control	Modo 2: Manejo de barco en entorno virtual
6	Monitor principal	Presentación de ventana principal del simulador
7	Palanca de mando	Control de turbinas

Universidad Escola Superior Politécnica del Litoral Logotipo	Facultad Facultad de Ingeniería Mecánica y Ciencia de la Producción	
	Revisado por PhD. Marcelo Fajardo	
Tipo de plano Diseño Mecánico		No. DWG DCM-001
Elaborado por Alexandra Imaicela y André Aguirre		Rev. 1
Título Consola de mando		Fecha de elab. 28/1/2024
		Escala 1:9



Universidad Esuela Superior Politécnica del Litoral Logotipo 	Facultad	Facultad de Ingeniería Mecánica y Ciencias de la Producción
	Tipo de plano	Diseño Mecánico
	Revisado por	PhD. Marcelo Fajardo
	Elaborado por	Alexandra Imaicela y André Aguirre
	No. DWG	DWG-101
	Título	Medidas de consola de mando
	Rev.	1
	Fecha de elab.	28/01/2024
	Escales	1:9



Universidad Escola Superior Politécnica del Litoral Logotipo	Facultad Facultad de Ingeniería Mecánica y Ciencias de la Producción
	Aprobado por PhD. Marcelo Fajardo
	DWG No. DWG-102
Tipo de plano Diseño mecánico	Rev. 1
Elaborado por Alexandra Imaicela y André Aguirre	Fecha de elab. 28/01/2024
Título Plano de implantación	Escala 1:18

Apéndice G

Fichas técnicas de los componentes industriales

En este apéndice se presentan las fichas técnicas de las páginas más relevantes de cada uno de los componentes electrónicos; es decir, se presentarán las especificaciones de los selectores de dos y tres posiciones, la palanca de mando, el volante y el joystick.

Figura G.1

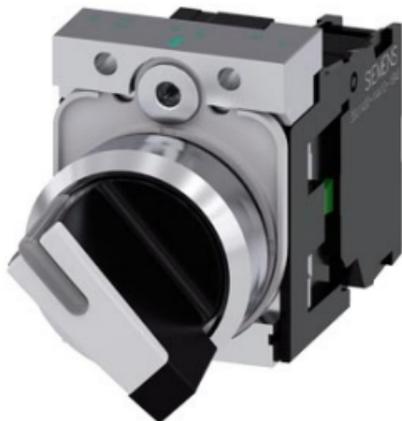
Ficha técnica del selector dos posiciones: circuito interno

SIEMENS

Hoja de datos

3SU1150-2BF60-1BA0

selector, iluminable, 22 mm, redondo, metal, brillante, blanco,
 selector, corto, 2 posiciones O-I, sostenido, ángulo de maniobra 90°,
 10:30h/13:30 h, con soporte, 1 NA, borne de tornillo



Nombre comercial del producto	SIRIUS ACT
Designación del producto	Selector
Tipo de producto	aparato completo
Denominación del tipo de producto	3SU1
Línea de productos	Metal, brillante, 22 mm
Referencia del fabricante	<ul style="list-style-type: none"> • 1 del módulo de contacto suministrado 3SU1400-1AA10-1BA0 • del soporte suministrado 3SU1550-0AA10-0AA0 • del actuador suministrado 3SU1052-2BF60-0AA0

Carcasa

Número de puntos de mando	1
----------------------------------	---

Actuador

Tipo de elemento de accionamiento	Selector, corto
Modo de funcionamiento del actuador	posiciones permanentes, 90° (10:30h/13:30h)
Ampliación del producto opcional Fuente luminosa	Sí
Color del actuador	blanco
Material del actuador	plástico
Forma del actuador	redondo
Diámetro exterior del actuador	32,3 mm

Nota. Componente fabricado por Siemens. Página 1.

Figura G.2

Ficha técnica del selector dos posiciones: circuito interno

Número de módulos de contacto	1
Número de posiciones de maniobra	2
Ángulo de maniobra	90°
• hacia la derecha	
Anillo frontal	
Componente del producto anillo frontal	Sí
Tipo de anillo frontal	standard
Material del anillo frontal	Metal brillante
Color del anillo frontal	plateado
Soporte	
Material del soporte	Metal
Indicación	
Número de módulos LED	0
Datos técnicos generales	
Función del producto	No
• apertura positiva	
Componente del producto	No
• Fuente luminosa	
Tensión de aislamiento	500 V
• valor asignado	
Grado de contaminación	3
Tipo de corriente	AC/DC
• de la tensión de empleo	
Resistencia a tensión de choque valor asignado	6 kV
Grado de protección IP	IP66, IP67, IP69(IP69K)
• del borne de conexión	IP20
Grado de protección NEMA	NEMA 1, 2, 3, 3R, 4, 4X, 12, 13
Resistencia a choques	Semiperíodo senoidal 50 g / 11 ms
• según IEC 60068-2-27	
Resistencia a vibraciones	10 ... 500 Hz: 5g
• según IEC 60068-2-6	
Frecuencia de maniobra máx.	1 800 1/h
Vida útil mecánica (ciclos de maniobra)	1 000 000
• típico	
Vida útil eléctrica (ciclos de maniobra)	10 000 000
• típico	
Corriente térmica	10 A
Designaciones de referencia según IEC 81346-2:2009	S
Designaciones de referencia según EN 61346-2	S

Nota. Componente fabricado por Siemens. Página 2.

Figura G.3

Ficha técnica del selector dos posiciones: circuito interno

Corriente permanente del interruptor automático curva C	10 A; para una intensidad de cortocircuito inferior a 400 A
Corriente permanente del cartucho fusibles DIAZED rápido	10 A
Corriente permanente del cartucho fusible DIAZED gG	10 A
Tensión de empleo	
<ul style="list-style-type: none"> • con AC <ul style="list-style-type: none"> — con 50 Hz valor asignado — con 60 Hz valor asignado • con DC <ul style="list-style-type: none"> — valor asignado 	5 ... 500 V 5 ... 500 V 5 ... 500 V
Electrónica de potencia	
Confiabilidad de contacto	Una maniobra errónea cada 100 millones (17 V, 5 mA), una maniobra errónea cada 10 millones (5 V, 1 mA)
Circuito de corriente secundaria	
Tipo de contacto de los contactos auxiliares	Aleación de plata
Número de contactos NC para contactos auxiliares	0
Número de contactos NA para contactos auxiliares	1
Conexiones/ Bornes	
Tipo de conexión eléctrica	
<ul style="list-style-type: none"> • de módulos y accesorios 	Bornes de tornillo
Tipo de secciones de conductor conectables	
<ul style="list-style-type: none"> • monofilar con preparación de los extremos de cable • monofilar sin preparación de extremos de cable • alma flexible con preparación de los extremos de cable • alma flexible sin preparación de extremos de cable • con cables AWG 	2x (0,5 ... 0,75 mm ²) 2x (1,0 ... 1,5 mm ²) 2x (0,5 ... 1,5 mm ²) 2x (1,0 ... 1,5 mm ²) 2x (18 ... 14)
Par de apriete de los tornillos en el soporte	1 ... 1,2 N·m
Par de apriete	
<ul style="list-style-type: none"> • con bornes de tornillo 	0,8 ... 0,9 N·m
Seguridad	
Valor B10	
<ul style="list-style-type: none"> • con alta tasa de demanda según SN 31920 	100 000
Cuota de defectos peligrosos	
<ul style="list-style-type: none"> • con baja tasa de demanda según SN 31920 • con alta tasa de demanda según SN 31920 	20 % 20 %
Tasa de fallos [valor FIT]	
<ul style="list-style-type: none"> • con baja tasa de demanda según SN 31920 	100 FIT

Nota. Componente fabricado por Siemens. Página 3.

Figura G.4

Ficha técnica del selector dos posiciones: circuito interno

Valor T1 para intervalo entre pruebas o vida útil según IEC 61508	20 y
Condiciones ambiente	
Temperatura ambiente	
• durante el funcionamiento	-25 ... +70 °C
• durante el almacenamiento	-40 ... +80 °C
Categoría medioambiental durante el funcionamiento según IEC 60721	3M6, 3S2, 3B2, 3C3, 3K6 (con humedad relativa del aire de 10 ... 95 %, no se permite condensación en servicio en ninguno de los aparatos por detrás del panel frontal)
Instalación/ fijación/ dimensiones	
Tipo de fijación	
• de módulos y accesorios	Fijación en placa frontal
Altura	40 mm
Anchura	32,3 mm
Forma del recorte para montaje	redondo
Díámetro de montaje	22,3 mm
Tolerancia positiva del diámetro de montaje	0,4 mm
Altura de montaje	28,8 mm
Ancho de montaje	32,3 mm
Profundidad de montaje	49,7 mm

Nota. Componente fabricado por Siemens. Página 4.

Figura G.5

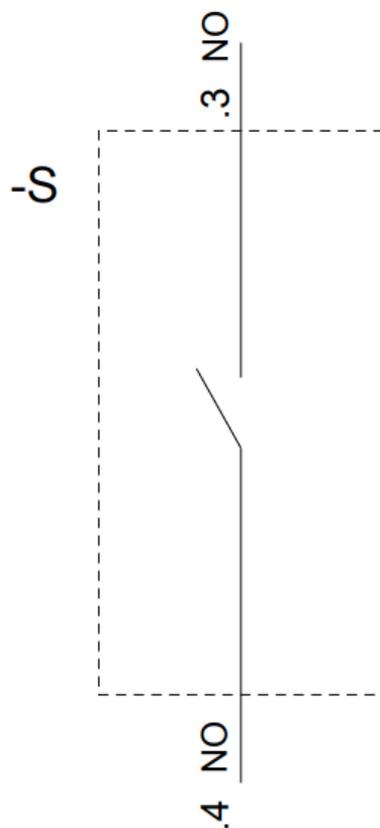
Ficha técnica del selector dos posiciones: diseño mecánico



Nota. Componente fabricado por Siemens. Página 5.

Figura G.6

Ficha técnica del selector dos posiciones: diseño eléctrico



Última modificación:

13/01/2020 

Nota. Componente fabricado por Siemens. Página 6.

Figura G.7

Ficha técnica del selector tres posiciones: circuito interno

SIEMENS**Hoja de datos****3SU1130-2BL60-1NA0**

selector, iluminable, 22 mm, redondo, plástico con anillo frontal de metal, blanco, selector, corto, 3 posiciones I-O-II, sostenido, ángulo de conmutación 2x45°, 10:30h/12h/13:30 h, con soporte, 1 NA, 1 NA, borne de tornillo

nombre comercial del producto	SIRIUS ACT
designación del producto	Selector
tipo de producto	aparato completo
denominación del tipo de producto	3SU1
línea de productos	Plástico con anillo frontal de metal, mate, 22 mm
referencia del fabricante	
<ul style="list-style-type: none"> del módulo de contacto suministrado en la posición 1 del módulo de contacto suministrado en la posición 2 del soporte suministrado del actuador suministrado 	3SU1400-1AA10-1BA0 3SU1400-1AA10-1BA0 3SU1550-0AA10-0AA0 3SU1032-2BL60-0AA0
Carcasa	
número de puntos de mando	1
Actuador	
tipo de elemento de accionamiento	Selector, corto
modo de funcionamiento del actuador	posiciones permanentes, 2x45° (10:30h/12h/13:30h)
ampliación del producto opcional fuente luminosa	Si
color del actuador	blanco
materia del actuador	plástico
forma del actuador	redondo
diámetro exterior del actuador	32,3 mm
número de módulos de contacto	2
número de posiciones de maniobra	3
ángulo de maniobra	
<ul style="list-style-type: none"> hacia la derecha hacia la izquierda 	45° 45°
Anillo frontal	
componente del producto anillo frontal	Si
tipo de anillo frontal	standard
materia del anillo frontal	Metal mate
color del anillo frontal	gris arena
Soporte	
materia del soporte	Plástico
Indicación	
número de módulos LED	0
Datos técnicos generales	
función del producto apertura positiva	No
componente del producto fuente luminosa	No
tensión de aislamiento valor asignado	500 V
grado de contaminación	3
tipo de corriente de la tensión de empleo	AC/DC

Nota. Componente fabricado por Siemens. Página 1.

Figura G.8

Ficha técnica del selector tres posiciones: circuito interno

resistencia a tensión de choque valor asignado	6 kV
grado de protección IP	IP66, IP67, IP69(IP69K)
• del borne de conexión	IP20
grado de protección NEMA	1, 2, 3, 3R, 4, 4X, 12, 13
resistencia a choques	
• según IEC 60068-2-27	onda semi-sinusoidal 15g / 11 ms
• para aplicaciones ferroviarias según EN 61373	categoría 1, clase B
resistencia a vibraciones	
• según IEC 60068-2-6	10 ... 500 Hz: 5g
• para aplicaciones ferroviarias según EN 61373	categoría 1, clase B
frecuencia de maniobra máx.	1 800 1/h
vida útil mecánica (ciclos de maniobra) típico	1 000 000
durabilidad eléctrica (ciclos de maniobra) típico	10 000 000
corriente térmica	10 A
designaciones de referencia según IEC 81346-2:2009	S
corriente permanente del interruptor automático curva C	10 A; para una corriente de cortocircuito inferior a 400 A
corriente permanente del cartucho fusibles DIAZED rápido	10 A
corriente permanente del cartucho fusible DIAZED gG	10 A
Directiva RoHS (fecha)	10/01/2014
tensión de empleo	
• con AC	
— con 50 Hz valor asignado	5 ... 500 V
— con 60 Hz valor asignado	5 ... 500 V
• con DC valor asignado	5 ... 500 V
Electrónica de potencia	
confiabilidad de contacto	Una maniobra errónea cada 100 millones (17 V, 5 mA), una maniobra errónea cada 10 millones (5 V, 1 mA)
Circuito de corriente secundario	
tipo de contacto de los contactos auxiliares	Aleación de plata
número de contactos NC para contactos auxiliares	0
número de contactos NA para contactos auxiliares	2
Conexiones/ Borneos	
tipo de conexión eléctrica	conexión por tornillo
• de módulos y accesorios	Borneos de tornillo
tipo de secciones de conductor conectables	
• monofilar con preparación de los extremos de cable	2x (0,5 ... 0,75 mm ²)
• monofilar sin preparación de extremos de cable	2x (1,0 ... 1,5 mm ²)
• alma flexible con preparación de los extremos de cable	2x (0,5 ... 1,5 mm ²)
• alma flexible sin preparación de extremos de cable	2x (1,0 ... 1,5 mm ²)
• con cables AWG	2x (18 ... 14)
par de apriete de los tornillos en el soporte	1 ... 1,2 N·m
par de apriete con bornes de tornillo	0,8 ... 0,9 N·m
Seguridad	
cuota de defectos peligrosos	
• con baja tasa de demanda según SN 31920	20 %
• con alta tasa de demanda según SN 31920	20 %
tasa de fallos [valor FIT] con baja tasa de demanda según SN 31920	100 FIT
valor B10 con alta tasa de demanda según SN 31920	300 000
IEC 62061	
valor T1 para intervalo entre pruebas o vida útil según IEC 61508	20 a
Condiciones ambiente	
temperatura ambiente	
• durante el funcionamiento	-25 ... +70 °C
• durante el almacenamiento	-40 ... +80 °C
categoría medioambiental durante el funcionamiento según IEC 60721	3M6, 3S2, 3B2, 3C3, 3K6 (con humedad relativa del aire de 10 ... 95 %, no se permite condensación en servicio en ninguno de los aparatos por detrás del panel frontal)
Instalación/ fijación/ dimensiones	
tipo de fijación	
• de módulos y accesorios	Fijación en placa frontal

Nota. Componente fabricado por Siemens. Página 2.

Figura G.9

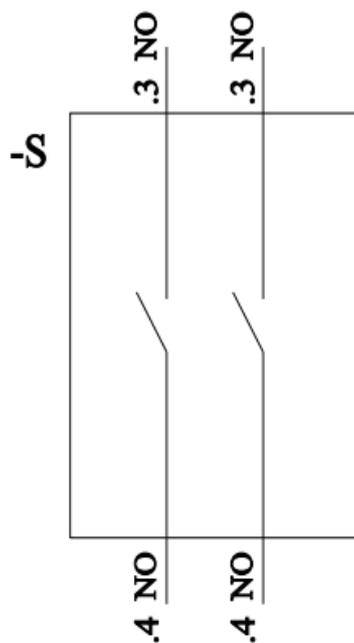
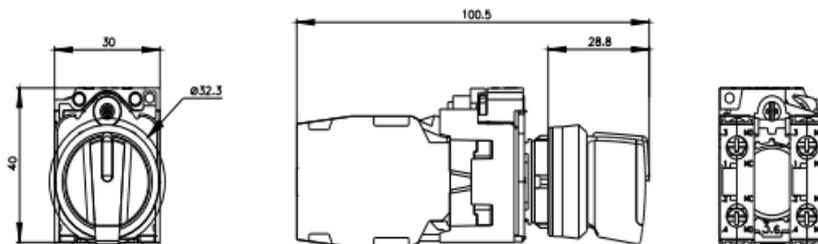
Ficha técnica del selector tres posiciones: circuito interno

altura	40 mm				
anchura	32,3 mm				
forma del recorte para montaje	redondo				
diámetro de montaje	22,3 mm				
tolerancia positiva del diámetro de montaje	0,4 mm				
altura de montaje	28,8 mm				
ancho de montaje	32,3 mm				
profundidad de montaje	49,7 mm				
Homologaciones Certificados					
General Product Approval					
			Confirmation		
General Product Approval	Test Certificates	Marine / Shipping			
	Special Test Certificate	Type Test Certificates/Test Report			
Marine / Shipping	other				
	Confirmation				

Nota. Componente fabricado por Siemens. Página 3.

Figura G.10

Ficha técnica del selector tres posiciones: diseño mecánico y eléctrico



Última modificación:

26/1/2022

3SU11302BL601NA0
Página 4/5

31/1/2024

Sujeto a cambios
© Copyright Siemens

Nota. Componente fabricado por Siemens. Página 4.

Figura G.11

Ficha técnica del joystick tipo JP50

JP50

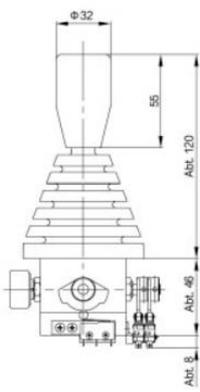
JP50 系列操纵杆
JP50 Series Joystick



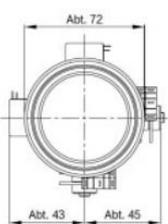
JP50-11-11-S12

- ◆ 中等尺寸操纵杆、采用进口电位器
Medium-sized joystick, imported potentiometer incorporated
- ◆ 功能定制性强
Strong function customization
- ◆ 可搭配模块实现总线输出
Available for combination with modules to realize Bus output

标准尺寸 / Standard Dimensions

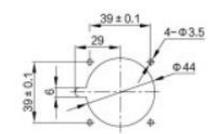


Side view dimensions: $\phi 32$, 55, Abt. 120, Abt. 46, Abt. 8



Front view dimensions: Abt. 72, Abt. 43, Abt. 45

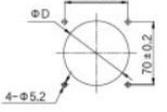
面板开孔尺寸 (不带安装面板)
Panel arrangements (without mounting panel)



Panel arrangement without panel: 39 ± 0.1 , 29, $4 - \phi 3.5$, $\phi 44$, 39 ± 0.1 , ϕ

建议面板厚度: 2mm-6mm
thickness of the panel: 2mm-6mm

面板开孔尺寸 (带标准面板)
Panel arrangements (With standard mounting panel)



Panel arrangement with panel: ϕD , 70 ± 0.2 , 70 ± 0.2 , $4 - \phi 5.2$

注: 尺寸D在产品选型时根据具体产品规格确定
Dimension D is determined by the product specification

Nota. Componente fabricado por SENTOP. Página 1.

Figura G.12

Ficha técnica del joystick tipo JP50

JP50

JP50 系列操纵杆
JP50 Series Joystick

↻ 一般性能 / General Specifications

操作角度 Operating angle	约 Abt. ±32° 全方位 Omni-directional
操作压力 (注) Operating force (note)	约 Approx. 3.5N~10N
预期机械寿命 Life expectancy	约Approx. 5,000,000
工作温度 Operating temperature	-20°C~+70°C
储存温度 Storage temperature	-20°C~+70°C
振动 Vibration	10~55Hz 98m/s²
冲击 Shock	294m/s²
质量 Mass	约 Approx. 460g
防护等级 Protection degree	IP65 (面板以上 Above panel)

注: 测力点距离安装面距离为115mm. Note The measuring point is 115 mm away from the subpanel for mounting.

↻ 电气性能 (电位器) / Electrical Specifications (Potentiometer)

阻值 Standard resistance value	10K±15%
独立线性公差 Independent linearity tolerance	±3%
输出平滑 Output smoothness	<输入电压 Against input voltage 0.2%
触点电阻变异 Contact resistance variation	<5%CRV
分辨率 Resolution	无穷小 Essentially infinite
功率 Power rating	0.2W
介质耐压 Dielectric strength	>1000MQ (500V D.C.)
绝缘电阻 Insulation resistance	1 min at (500VAC)
中心抽头角度 Center tap angle	约 Approx. 3° (41.5°)
复位精度 (无抽头) Repeatability of tolerance (no center tap)	约 Approx. ±2%
通过电刷最大电流 Wiper current	10µA ~1mA (Max. 1mA)
EMS EMS	10V/m (80MHz~1GHz 1KHz 80%)

↻ 电气性能 (开关) / Electrical Specifications (Switch)

电气性能 (开关) Electrical specifications (switch)	把手按钮开关 / Push button switch	微动开关 / Microswitch
容量 Rating	48V D.C. / 0.1A	125V A.C. / 5A
电气寿命 Electrical life expectancy	1,000,000	10,000
机械寿命 Mechanical life expectancy	-	10,000,000
绝缘电阻 Insulation resistance	>1000MQ (500V D.C.)	>100MQ (500V D.C.)
介质耐压 Dielectric strength	1min (1000V A.C.)	1min (1000V A.C.)

↻ 其他可选把手 / Other Optional Knobs

↻ 可选特殊规格 / Special Specifications Available

其他输出信号 Other output signals	CAN OPEN, USB信号, 无线遥控, 数字信号输出, 双路信号输出等 CAN OPEN, USB signal, wireless remote control, digital signal output, dual signal output etc.
特殊把手 Special knob	特殊定制把手, 带按钮开关把手 (可选带灯), 微动开关可选其他型号等 Customized knob, with push button switch (with light is optional), with other microswitches models, etc.

* 如有其他特殊要求请与贵商联系. For other special requirements, please contact with SIBO.
(在产品特殊规格时, 其一般机械电气性能和环境性能可能发生改变, 请预先向贵商确认. In case of the potentiometer with special specifications, the general mechanical, electrical specifications and environmental specifications may change. Please consult with SIBO in advance.)
(为了不断提高质量和设计, 所有内附的规格可能在没有事先通知的情况下进行改变. All details given in this catalog may be changed without prior notice in order to continuously improve quality and design of our products.)
版权归贵商所有, 盗用必究. All rights are reserved by SIBO company. Embecclement must be protected.



www.sakae.cn

Nota. Componente fabricado por SENTOP. Página 2.

Figura G.13

Ficha técnica del volante/timón

ENTRETENIMIENTO > **VOLANTES**
A40122000377
 Volante Potenciómetro 270°
 (V/270-P)

Industrias Lorenzo, S.A.

Documento Técnico

Propiedades

Fabricado con materiales y componentes de primera línea.

Diseño Robusto y Compacto.

Efecto realista Giro del Volante.

Giro de 270°.

Forzado posición central del Volante.

Disponible con 4 tipos diferentes de Aros.

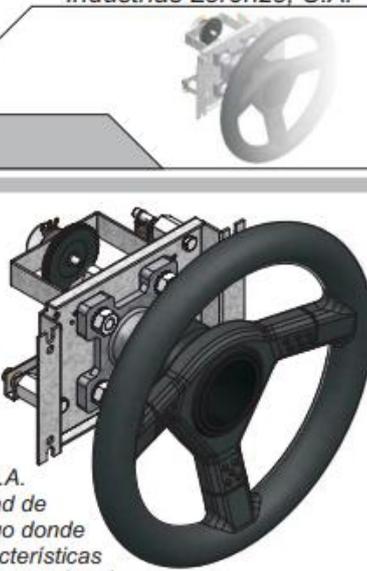
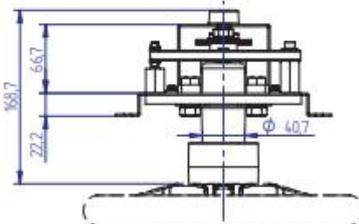
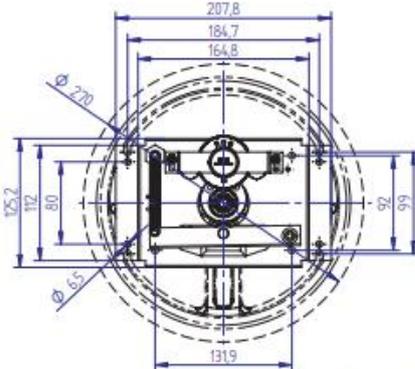
Aplicaciones

Para máquinas de juego de conducción.

Para máquinas que requieran un direccionamiento progresivo mediante Potenciómetro..

El Volante Potenciómetro **V/270-P** se puede montar en cualquier máquina de juego de **conducción de vehículos** y otros donde se requiera un **direccionamiento**.

(Industrias Lorenzo, S.A. tiene una gran variedad de volantes en su catálogo donde hay modelos con características diversas. Entre ellas los grados de giro y control de giro. Para más información vea <http://www.industrias-lorenzo.com>)

Propiedades Eléctricas	
POTENCIÓMETRO	
Resistencia	5k (Ohms)
Tolerancia Resistencia	±10%
Potencia	2W (Watts)
Vida	25.000 Ciclos

www.industrias-lorenzo.com

Español Rev:1.0.0

Contacto
 Telf: (+34) 936 47 34 50
 e-mail: sales@industrias-lorenzo.com
 Avda. Prat de la Riba, 84
 08849 Sant Climent de Llobregat
 BARCELONA (ESPAÑA)

Industrias Lorenzo, S.A.

-1-

Nota. Componente fabricado por Industrias Lorenzo, S.A. Página 1.

Figura G.14

Ficha técnica del volante/timón

ENTRETENIMIENTO > VOLANTES

A40122000377

Volante Potenciómetro 270°
(V/270-P)

Documento Técnico

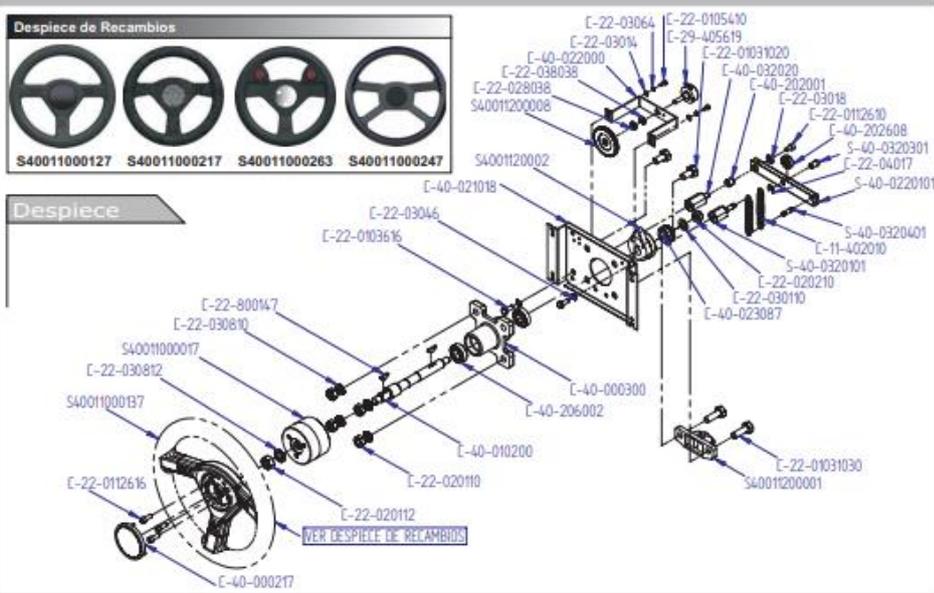
Industrias Lorenzo, S.A.



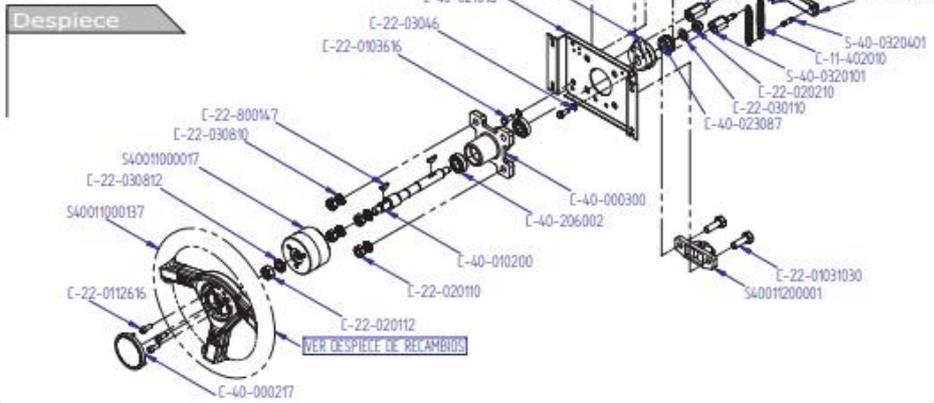
Despiece de Recambios



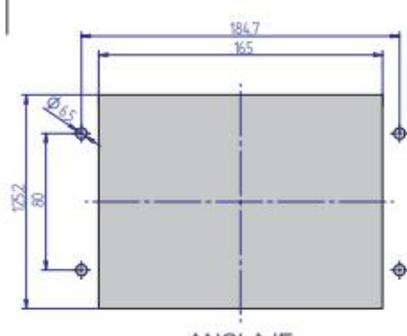
S40011000127 S40011000217 S40011000263 S40011000247



Despiece



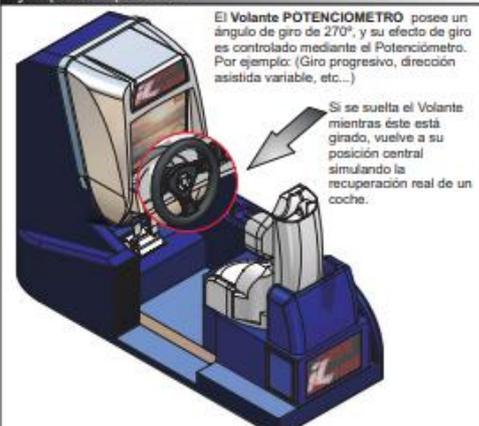
Panel de Montaje



ANCLAJE

NOTA:
Disponibles diferentes "Aros de Volante".

Ejemplo de aplicación



El Volante **POTENCIOMETRO** posee un ángulo de giro de 270°, y su efecto de giro es controlado mediante el Potenciómetro. Por ejemplo: (Giro progresivo, dirección asistida variable, etc...)

Si se suelta el Volante mientras éste está girado, vuelve a su posición central simulando la recuperación real de un coche.

NOTA: Bajo consulta y pedido se pueden realizar adaptaciones a sus necesidades.

www.industrias-lorenzo.com Español Rev:1.0.0

Contacto
Telf: (+34) 936 47 34 50
e-mail: sales@industrias-lorenzo.com
Avingda. Prat de la Ribba, 84
08849 Sant Climent de Llobregat
BARCELONA (ESPAÑA)



Industrias Lorenzo, S.A.

-2-

Nota. Componente fabricado por Industrias Lorenzo, S.A. Página 2.

Figura G.15

Ficha técnica del Arduino MEGA2560: circuito interno

**Arduino® MEGA 2560 Rev3**

Product Reference Manual
SKU: A000067

**Description**

Arduino® Mega 2560 is an exemplary development board dedicated for building extensive applications as compared to other maker boards by Arduino. The board accommodates the ATmega2560 microcontroller, which operates at a frequency of 16 MHz. The board contains 54 digital input/output pins, 16 analog inputs, 4 UARTs (hardware serial ports), a USB connection, a power jack, an ICSP header, and a reset button.

Target Areas

3D Printing, Robotics, Maker

Nota. Componente fabricado por Arduino. Página 1.

Figura G.16

Ficha técnica del Arduino MEGA2560: circuito interno



Arduino® MEGA 2560 Rev3

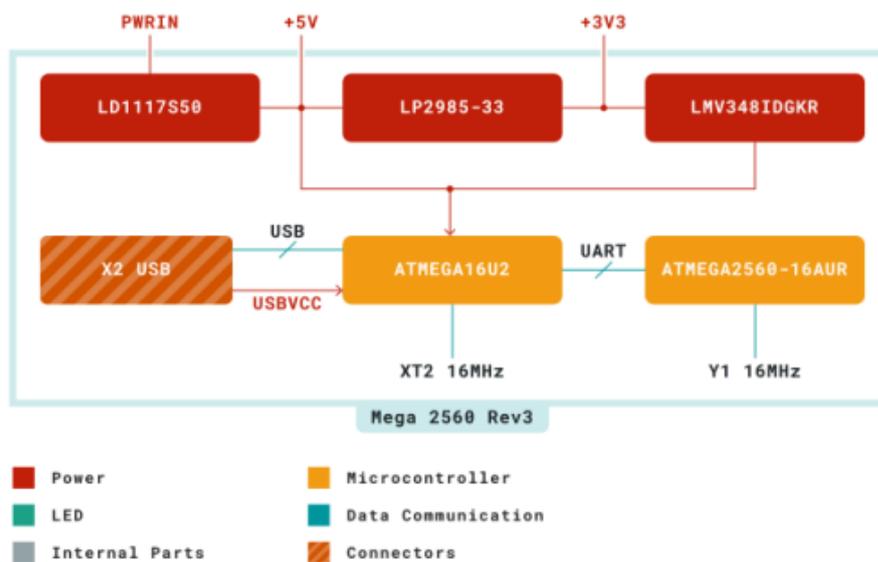
2 Ratings

2.1 Recommended Operating Conditions

Symbol	Description	Min	Typ	Max	Unit
V_{IN}	Input voltage from VIN pad / DC Jack	7	7.0	12	V
V_{USB}	Input voltage from USB connector	4.8	5.0	5.5	V
T_{OP}	Operating Temperature	-40	25	85	°C

3 Functional Overview

3.1 Block Diagram

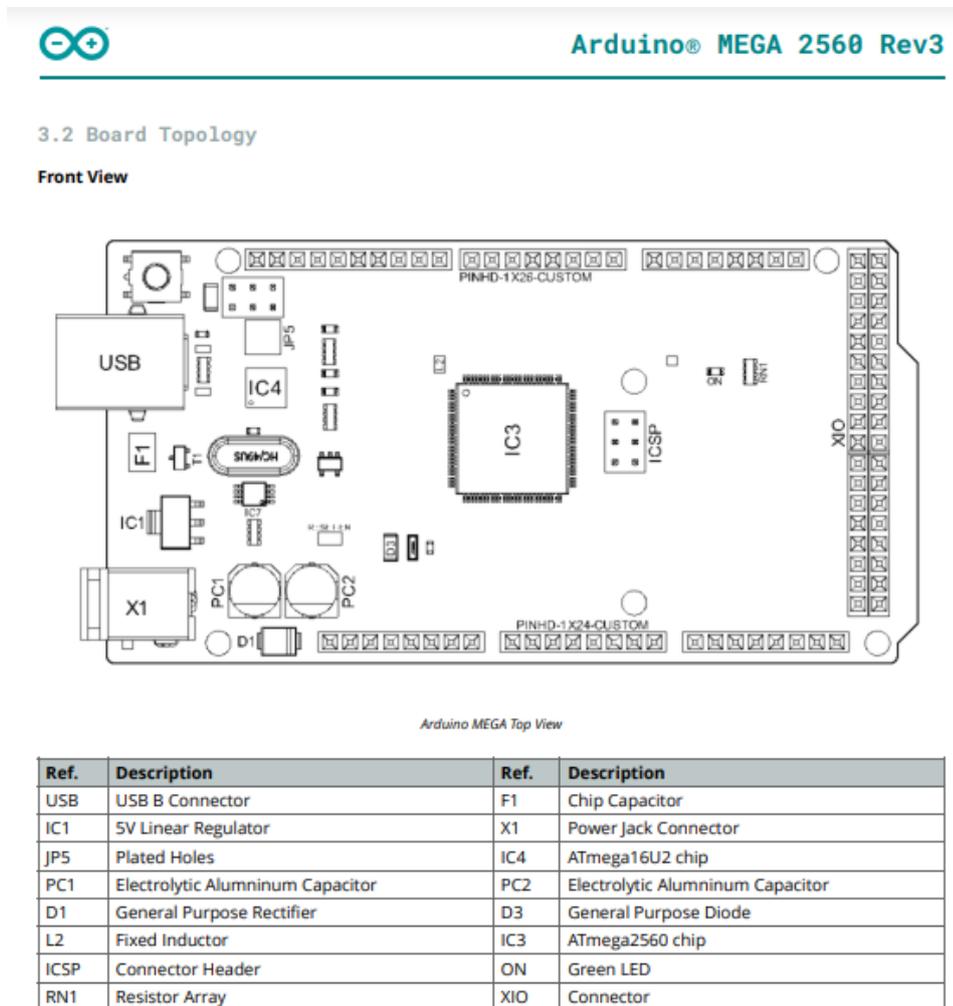


Arduino MEGA Block Diagram

Nota. Componente fabricado por Arduino. Página 5.

Figura G.17

Ficha técnica del Arduino MEGA2560: Topología de placa electrónica



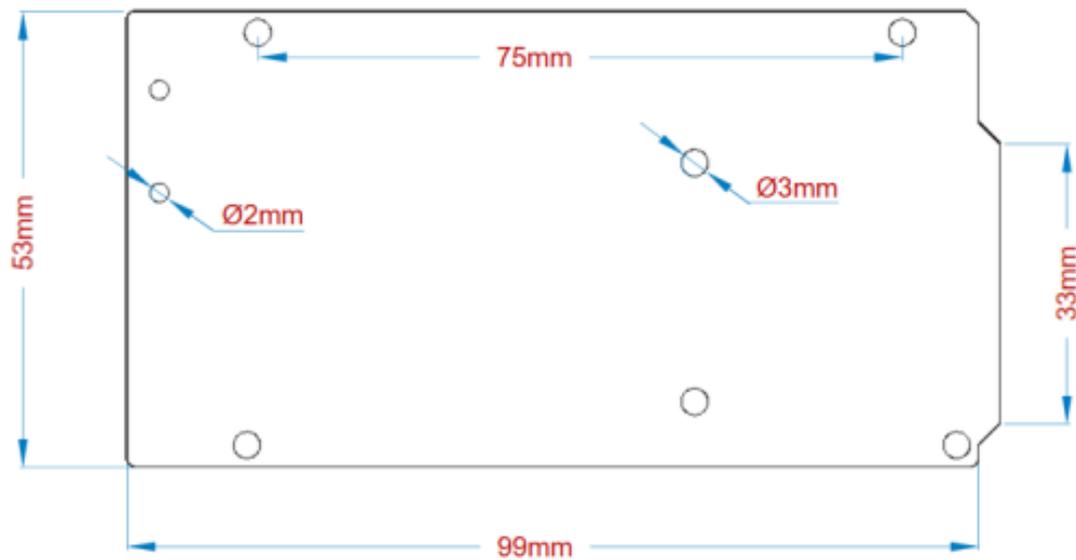
Nota. Componente fabricado por Arduino. Página 6.

Figura G.18

Ficha técnica del Arduino MEGA2560: diseño mecánico

6 Mechanical Information

6.1 Board Outline

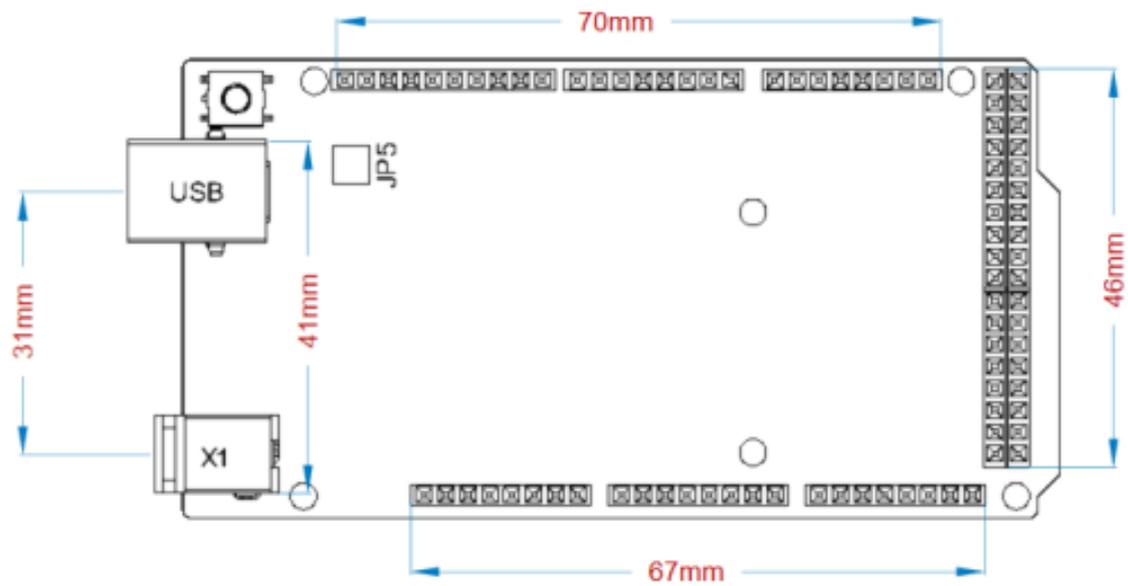


Nota. Componente fabricado por Arduino. Página 13.

Figura G.19

Ficha técnica del Arduino MEGA2560: diseño mecánico

6.2 Board Mount Holes



Nota. Componente fabricado por Arduino. Página 14.

Figura G.20

Ficha técnica de la palanca de mando

NORISYS 4 LS4 Control Lever System

- Single lever and double lever setups
- Several available scales, separated for both handles
- LED band for position indication of active lever for each handle
- Optional electrical shaft functionality for each handle with force feedback
- 2 separated CANbus interfaces (option)
(CAN1 can be configured as RS-232/RS-485 interface)
- 1 RS-485 interface (optional)
- 1 scale illumination input (dimmable)
- 2 digital inputs, galvanically isolated (optional)
- 2 analogue outputs 4 ... 20 mA (one for each handle, optional)
- Extended operating temperature range -25°C ... +70°C
- IP56 front side



Control lever system NORISYS4 LS4



Application range

The NORISTAR control lever system is designed for ship propulsion plant applications in accordance to marine certification requirements. The lever can be equipped in three levels, starting from a mechanical setup with potentiometric signal outputs, basic electronic equipment with analogue standard signal output 4 ... 20 mA for each handle and as full electric version with integrated data interface and optional electrical shaft system onboard.

Description

In relation to its area of application the lever can be equipped as single or double lever as well as control lever chain. The portfolio of standard and customer-specific scales matches a wide range of applications. Direct wiring of standard industrial signal cables is provided by 2.5 mm² terminal blocks. The design as a plug-and-play component in the basic and full electronic version requires no calibration handling on customer side. The full electronic version is equipped with a high performance ARM processor, which calculates the handle positions, controls the integrated LED band as well as the stepper motors of the optional electrical shaft system and powers the data interfaces. The integrated LED band for each handle is a precise visualisation to indicate the current position of the active control lever and to support the operator during control position transfer. An optional electrical shaft system provides automatic alignment of each handle according to the position of the active control lever in the network. The ESS option uses the existing network interconnection between all levers and the remote control system and requires no separate control hardware.

Interconnection

The full electronic version is equipped with several data interfaces as well as analogue standard signal outputs. The full electronic equipped control lever can be interconnected to an automation system via redundant or single CANbus as well as by using the integrated RS-485 interface with Modbus-RTU or NORISYS 4 ExtBus protocol. The electronic control lever can be used as gateway to add NORISYS 4 and NORISTAR 4 extension units to an automation system. All versions provide a signal output for each handle, positioning indication and dimming of the scale illumination. The data interfaces are short-circuit protected and 24 V protected.

Mechanical Versions

The mechanical design allows a setup of several application specific versions. The lever can be equipped as single and double handle. For main propulsion systems a base socket can be used to tend the device towards the operator. For thruster applications the control lever can be mounted rotated by 90°. The handle can be mounted according to application and user requirements. For similar propulsion plants it is possible to establish a control lever chain by connecting the control levers with a reversible mechanical linkage.

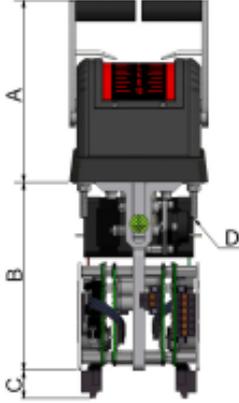
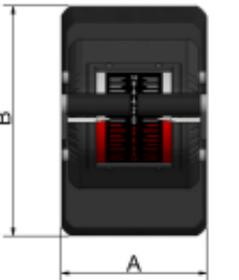
Nota. Componente fabricado por NORIS AUTOMATION. Página 1.

Figura G.21

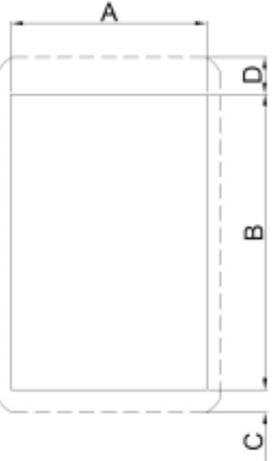
Ficha técnica de la palanca de mando: conexiones y dimensiones

Dimensions, connections and drawings

Device dimensions

	<p>Explanation to the left illustration (side view)</p> <p>A) Length 128 mm B) Length 132 mm C) Length 20 mm D) Thread M5, length 25 mm</p>
	<p>Explanation to the left illustration (above view)</p> <p>A) Length 96 mm B) Length 154 mm</p>

Desk cut-out

	<p>Explanation to the left illustration</p> <p>A) Length 84 mm B) Length 128 mm C) Length 10 mm D) Length 16 mm</p>
---	---

Nota. Componente fabricado por NORIS AUTOMATION. Página 2.

Figura G.21

Ficha técnica de la palanca de mando: conexiones internas

Technical data

Connection	
Supply voltage	U _{nom} 24 VDC, 18 ... 32 VDC
Current consumption	0.15 ... 1.5 A according to level of equipment
Reverse voltage protection	Integrated
Over voltage protection	Integrated

Interfaces	
CANbus (optional)	2 x
RS-485 (optional)	1 x, galvanically isolated
Electrical connections	Terminals for cable profile 2.5 mm ²

In-/Output	
Digital inputs	1 x Input, 1x Output, galvanically isolated
Illumination regulation input	For conventional 24 VDC PWM dimmer or 0 ... 24 VDC

Environmental influences	
Operating temperature	DIN IEC 60068-2-2 and DIN IEC 60068-2-1: -25°C ... +70°C
Climatic test	DIN IEC 60068-2-30 Db
Storage temperature	DIN IEC 60068-2: -40°C ... +85°C
Vibration resistance	DIN IEC 60068-2-6 Fc: ±1.0 mm @ 2 ... 13.2 Hz, ±0.7 g @ 13.2 ... 100 Hz
Degree of protection	DIN EN 60529: IP56 front side
ESD	IEC 61000-4-2: ± 6 kV/Contact Discharge; ± 8 kV/Air Discharge
HF-interference immunity	IEC 61000-6-2; IEC 61000-4-3, -4-4, -4-5, -4-6
Interference emission	IEC 61000-6-4; CISPR16-1, CISPR16-2, EMC 1

Mechanical dimensions	
Material	Enclosure: PUR, AlMg3
Mounting	Console mounting
Installation position	None
Dimensions	96 x 154 x 280 mm (152 mm under floor)
Weight	1.8 kg - 2.4 kg according to level of equipment

Other	
ESS	Optional electrical shaft system with separate 24 VDC power supply
Approvals	CE, BV, DNV GL, LR, NKK, KR

Nota. Componente fabricado por NORIS AUTOMATION. Página 3.

Figura G.22

Ficha técnica de monitor de consola de mando

Ficha Técnica	
Exhibición	
Diagonal de la pantalla	49,5 cm (19.5")
Resolución de la pantalla	1600 x 900 Píxeles
Relación de aspecto nativa	16:9
Tipo de pantalla	TN
Brillo de la pantalla (típico)	250 cd / m ²
Tiempo de respuesta	5 ms
Tipo HD	HD+
Tecnología de visualización	LED
Forma de la pantalla	Plana
Formatos gráficos soportados	1600 x 900
Razón de contraste (típica)	1000:1
Máxima velocidad de actualización	60 Hz
Ángulo de visión, horizontal	170°
Ángulo de visión, vertical	160°
Número de colores de la pantalla	16,78 millones de colores
Tipo de retroiluminación	W-LED
Superficie de la pantalla	Mate
Relación de aspecto	16:9
Relación de contraste (dinámico)	20000000:1
3D	✘
Tamaño de pixel	0,2712 x 0,2712 mm
Intervalo de escaneado horizontal	30 – 83 kHz
Puertos e Interfaces	
Conector USB incorporado	✘
Puerto DVI	✘
Salida de auriculares	✔
Cantidad de puertos VGA (D-Sub)	1
Número de puertos HDMI	1
Versión HDMI	1.4
Salidas para auriculares	1
Auriculares	3,5 mm
Micrófono, jack de entrada	✔

Nota. Componente fabricado por AOC. Información obtenida de internet.

Figura G.23*Ficha técnica de monitor de consola de mando*

Intervalo de escaneado vertical	50 – 76 Hz		
Tamaño visible, horizontal	43,4 cm		
Tamaño visible, vertical	23,6 cm		
sRGB	☑		
Cobertura sRGB (típica)	99%		
Profundidad de color	8 bit		
Cobertura Adobe RGB	73%		
Peso y dimensiones		Control de energía	
Ancho	14 cm	Clase de eficiencia de energía	A
Altura	36 cm	Consumo energético	14 W
Profundidad	52 cm	Consumo de energía (inactivo)	0,5 W
Peso	3.3 kg	Voltaje de entrada AC	100 – 240 V
		Frecuencia de entrada AC	50 – 60 Hz
		Valoración actual	1,5 A
		Adaptador de energía externo	☒
Condiciones ambientales		Empaquetado	
Intervalo de temperatura operativa	0 – 40 °C	Stand incluido	☑
Intervalo de temperatura de almacenaje	-25 – 55 °C	Cables incluidos	Corriente alterna, VGA
Intervalo de humedad relativa para funcionamiento	10 – 85%	Longitud del cable VGA	1,8 m
Intervalo de humedad relativa durante almacenaje	5 – 93%	Ancho del paquete	521 mm
Altitud de funcionamiento	0 – 5000 m	Profundidad del paquete	140 mm
Altitud no operativa	5000 – 12192 m	Altura del paquete	368 mm
		Peso del paquete	7,63 kg
		Tipo de embalaje	Caja
Ergonomía		Otras características	
montaje VESA	☑	Longitud del cable	1,8 m
Ajustes de altura	☒	Interruptor de encendido/apagado	☑

Nota. Componente fabricado por AOC. Información obtenida de internet.

Figura G.24*Ficha técnica de monitor de consola de mando*

Interfaz de panel de montaje	100 x 100 mm	integrado	
Ajuste de la inclinación	✔	Sintonizador de TV integrado	✘
Ángulo de inclinación	-3,5 – 21,5°	Tipo de grosor del bisel embellecedor	Normal
Exhibición en pantalla (OSD)	✔	Tiempo medio entre fallos	50000 h
Idiomas OSD	Chino simplificado, Chino tradicional, Checo, Alemán, Holandés, Inglés, Español, Finés, Francés, Italiano, Japonés, Coreano, Polaco, Portugués, Ruso, Sueco		
Número de lenguajes OSD	16		
Conectar y usar (Plug and Play)	✔		
Diseño		Desempeño	
Posicionamiento de mercado	Negocios	Tecnología Flicker free (reduce el parpadeo de la pantalla)	✔
Color del producto	Negro	Tecnología Low Blue Light (baja luz azul)	✔
Certificado Energy Star	✔		
Conformidad EPEAT	Silver		
Bisel frontal	Negro		
Soporte desmontable	✔		
Color de pies	Negro		
Libre de metales pesados	Hg (mercurio), Pb (plomo)		
Acorde RoHS	✔		
Multimedia			
Altavoces incorporados	✘		
Cámara incorporada	✘		

Nota. Componente fabricado por AOC. Información obtenida de internet.

Figura G.25*Ficha técnica de monitor de consola de mando*

Interfaz de panel de montaje	100 x 100 mm	integrado	
Ajuste de la inclinación	✔	Sintonizador de TV integrado	✘
Ángulo de inclinación	-3,5 – 21,5°	Tipo de grosor del bisel embellecedor	Normal
Exhibición en pantalla (OSD)	✔	Tiempo medio entre fallos	50000 h
Idiomas OSD	Chino simplificado, Chino tradicional, Checo, Alemán, Holandés, Inglés, Español, Finés, Francés, Italiano, Japonés, Coreano, Polaco, Portugués, Ruso, Sueco		
Número de lenguajes OSD	16		
Conectar y usar (Plug and Play)	✔		
Diseño		Desempeño	
Posicionamiento de mercado	Negocios	Tecnología Flicker free (reduce el parpadeo de la pantalla)	✔
Color del producto	Negro	Tecnología Low Blue Light (baja luz azul)	✔
Certificado Energy Star	✔		
Conformidad EPEAT	Silver		
Bisel frontal	Negro		
Soporte desmontable	✔		
Color de pies	Negro		
Libre de metales pesados	Hg (mercurio), Pb (plomo)		
Acorde RoHS	✔		
Multimedia			
Altavoces incorporados	✘		
Cámara incorporada	✘		

Nota. Componente fabricado por AOC. Información obtenida de internet.