

Centro Universitario de la Defensa en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Monitorización de variables de influencia en operación de buques para su modelado y visualización

Grado en Ingeniería Mecánica

ALUMNO: Francisco Antón Antón

DIRECTORES: Pedro Jesús Carrasco Pena

Gerardo González-Cela Echevarría

CURSO ACADÉMICO: 2022-2023

Universidade Vigo



Centro Universitario de la Defensa
en la Escuela Naval Militar

TRABAJO FIN DE GRADO

Monitorización de variables de influencia en operación de buques para su modelado y visualización

Grado en Ingeniería Mecánica

Intensificación en Tecnología Naval

Cuerpo General

Universidade Vigo

RESUMEN

La propiedad del dato y su tratamiento son una de las grandes preocupaciones de la industria de cuarta generación (i4.0) como así lo es también para la Armada. El auge de los gemelos digitales ha concienciado a diversos sectores sobre la necesidad de una herramienta de control y simulación de las plataformas, equipos y sistemas, existiendo numerosas implantaciones para lograr estos objetivos. El presente trabajo explica la concepción y diseño de una aplicación de análisis de datos con visualización interactiva para el tratamiento de información relativa a buques de la Armada, basando estos modelos en el estudio de variables en las Lanchas de Instrucción y buques de escolta. La metodología empleada para su desarrollo comprende las técnicas de Inteligencia Artificial, programación en Pyhton y la explotación del motor gráfico Unreal Engine, siendo este último la herramienta con la que se ha creado una ventana de visualización inmersiva, con el replicado del interior de las lanchas con técnicas de ingeniería inversa. Las técnicas de minería de datos empleadas hacen uso de las herramientas que Machine Learning proporciona para tratar con entornos de Big Data a partir de los cuales sea posible obtener un modelo para cada condición operativa de la plataforma.

PALABRAS CLAVE

Bases de datos, Gemelo Digital, modelado de buques, Python, Minería de Datos

AGRADECIMIENTOS

En primer lugar, agradecer a mis tutores, el TN Pedro Carrasco Pena y el CF Gerardo González-Cela Echevarría, por marcar la derrota a seguir a lo largo de este trabajo, así como por su interés y confianza a la hora de acceder a dirigir este proyecto.

A los colaboradores, fuentes de información experta. Al TN Costal Camiña y al CF Touza Gil, jefes del CESADAR periférico de Ferrol y del Centro de Inteligencia Artificial de la Armada respectivamente.

Al profesor Carlos Casqueiro Placer, por su tiempo y esfuerzo como apoyo en la fase de modelado.

Al Jefe de Mantenimiento del Arsenal de Ferrol, por su apoyo incondicional para marcar el paso en este trabajo y en todo lo que hiciera falta a lo largo de los años, gracias, papá.

A mi madre, a mi mayor soporte, por estar siempre al pie del cañón y aguantar en los momentos buenos, y los no tan buenos.

CONTENIDO

Contenido 1
Índice de Figuras 4
Índice de Tablas 6
1 Introducción y objetivos 7
1.1 Motivación 7
1.2 Objetivos 8
1.3 Metodología 8
1.4 Estructura de la memoria 9
2 Estado del arte 11
2.1 Contexto en la Armada 11
2.2 Sistema Integrado de Control de Plataforma 12
2.3 Centro de Supervisión y Análisis de Datos de la Armada 13
2.3.1 Automatización de Tareas de Vigilancia y Análisis (ATAVIA) 13
2.3.2 Sostenimiento Predictivo basado en Redes Neuronales (SOPRENE) 14
2.4 La industria 4.0 14
2.5 El gemelo digital en la Armada 16
2.6 Modelos de operación de Buques, el MMG 18
2.7 Software empleado en gemelos digitales 19
2.7.1 Visualización en 3D 19
2.7.2 Simulación 19
2.8 Inteligencia Artificial 20
2.9 Machine Learning (ML) 20
2.9.1 Fases del proceso de aprendizaje 21
2.9.2 Técnicas de ML aplicadas a la regresión 24
2.9.2.3 Support Vector Machines 25
2.9.2.4 Árboles de decisión 26
2.9.2.5 Random forests 27
2.9.2.6 Redes neuronales 28
2.9.2.7 Deep Learning 29
2.10 Lanchas Rodman 66 29
2.10.1 Sistema propulsor 30
2.11 Trabajos previos en el ámbito de estudio 31
2.11.1 Training Boat Digital Twin 31
2.11.2 El gemelo digital como base tecnológica para el desarrollo logístico de la Armada 4.0 31

2.11.3 Ghenova....31
3 Desarrollo del TFG....33
3.1 Generalidades....33
3.2 Reparto temporal de Tareas....33
3.2.1 Planificación....33
3.2.2 Fuentes de información....34
3.3 Descripción del entorno de trabajo....35
3.3.1 Miscrosoft Excel....35
3.3.2 Python....36
3.3.2.2 Jupyter Lab....37
3.3.2.3 Librerías de Python....37
3.3.3 Unreal Engine 5....38
3.4 Gestión de riesgos....38
3.5 Minería de datos y análisis....39
3.5.1 Selección de variables....39
3.5.2 Modelo inicial en base a buques de guerra sensorizados....40
3.5.3 Toma de datos en LLII y tratamiento....41
3.5.4 Análisis de los datos....42
3.6 Desarrollo de ventana de visualización del programa con Python....47
3.6.1 Streamlit....47
3.6.1.2 Página Data....50
3.6.2 Poka Yoke....56
3.7 Modelado 3D del interior de la LLII....56
3.7.1 Escaneo LIDAR y óptico....57
3.8 Entorno inmersivo con Unreal Engine 5....60
3.8.1 Generalidades....60
3.8.2 Creación del terreno....60
3.8.3 Importación de los modelo 3D....62
3.8.4 Importación de datos de buques....63
3.8.5 Desarrollo de los Widgets de visualización del MPD....64
3.8.6 Interacción modelo-Widget....65
3.8.7 Desarrollo del programa final....67
4 Resultados , validación y pruebas realizadas....69
4.1 Generalidades....69
4.2 Resultados de ML....69
4.2.1 Aplicabilidad de los modelos....69

4.2.1.1 Revoluciones frente a velocidad 70
4.2.1.2 Revoluciones frente a consumo de combustible 73
4.2.1.3 Revoluciones a presión del aceite de lubricación 78
4.3 Explotación de la aplicación 81
4.3.1 Grado de usabilidad 81
4.3.2 Guía de empleo 83
5 Conclusiones y líneas futuras 85
5.1 Conclusiones del trabajo 85
5.1.1 Grado de satisfacción de los objetivos iniciales 85
5.2 Líneas académicas 85
5.2.1 Mejoras 85
5.2.2 Ciclo de Vida del software 86
5.2.3 Explotación del proyecto actual en el ámbito académico 87
5.3 Líneas industriales 87
5.4 Líneas específicas del Ministerio de Defensa 88
6 Bibliografía 89
Anexo I: Implicaciones Sociales, Económicas y Ambientales 93
Anexo II: Reflexiones Éticas y Sociales 101
Anexo III: Glosario de siglas, Acrónimos y abreviaturas contenidas en el trabajo 103
Anexo IV: Planificación temporal del trabajo 105
Anexo V: Informe Leica Cyclone Register 360 PLUS 109
Anexo VI: Código Fuente Página Home 111
Anexo VII: Código Fuente Página Data 113
Anexo VIII: Código Fuente Página Visualización 131
Anexo IX: Código Fuente Página About 133
Anexo X: Blueprints UE5 widget Marine Power Display [er (1) y br(2)] 135
Anexo XI: Blueprints UE5 Interacción 137
Anexo XII: Documentación operativa para la toma de datos a bordo de las LLII 139

ÍNDICE DE FIGURAS

Figura 1-1. Organigrama general de dependencias de los datos de las plataformas de la Armada ... 8
Figura 2-1. Visión de la JAL 4.0 ... 12
Figura 2-2. Esquema de una ventana de visualización del SICP ... 13
Figura 2-3. Escala de tiempos de las diferentes revoluciones industriales ... 15
Figura 2-4.Concepto de astillero inteligente con fragata en grada en Ferrol ... 16
Figura 2-5. Bloques del modelo MMG ... 18
Figura 2-6. Diagrama de flujo del proceso de aprendizaje ... 21
Figura 2-7. Coeficientes de correlación y tipos de correlación entre 2 variables ... 22
Figura 2-8. Ejemplos de resultados en la validación de un modelo ... 24
Figura 2-9. Ejemplo del empleo de SVM para clasificación de buques tipo FF ... 25
Figura 2-10. Árbol de decisión buques ARFER ... 26
Figura 2-11. Técnica Random Forests ... 27
Figura 2-12. Función logística ... 29
Figura 2-13. Tecnologías dentro de ML ... 29
Figura 3-1. Interfaz MS Excel y Visual Basic ... 36
Figura 3-2. Interfaz de Jupyter Lab ... 37
Figura 3-3. Interfaz de la extensión Jupyter Notebook en Visual Studio Code ... 37
Figura 3-4. Entorno Unreal Engine ... 38
Figura 3-5. Resultados de la ampliación de datos con VBA. ... 42
Figura 3-6. Representación del filtrado por cuartiles. ... 46
Figura 3-7. Página “Home” ... 48
Figura 3-8. Distribución de los coeficientes de correlación ... 49
Figura 3-9. Esquema de diseño de la página "Data" ... 51
Figura 3-10. Zona "A". ... 52
Figura 3-11. Zona "B". ... 52
Figura 3-12. Zona "C". ... 53
Figura 3-13. Zona "D". ... 54
Figura 3-14. Página "Data" vista general. ... 54
Figura 3-15. Página "Visualization". ... 55
Figura 3-16. Página "About". ... 55
Figura 3-17. Solicitud de parámetros de la plataforma Poka Yoke en la ventana de visualización 56
Figura 3-18. Esquema del emplazamiento del sensor Leica RTC 360 en el puente de la A-121. ... 58
Figura 3-19. Resultados tratados por Cyclone 360 (izq) y exportados a Geomagic (der). ... 58
Figura 3-20. Mallas obtenidas con digitalización óptica. ... 60

Figura 3-21. Elaboración del terreno en UE con heightmaps 61
Figura 3-22. Blueprint textura terreno. 62
Figura 3-23. Modelo generado con SketchUp 62
Figura 3-24. Explicación tipo de colisión empleada 63
Figura 3-25. Estructura de un *Dataframe* 64
Figura 3-26. *Dataframe* importado a Unreal Engine con estructura definida 64
Figura 3-27. Interfaz de Widget en Unreal Engine 65
Figura 3-28. Blueprint función "get text" 65
Figura 3-29. Blueprint colisión con MPD de babor 66
Figura 3-30. Función *Row Index* y diagrama de Petri 67
Figura 4-1. Incertidumbres en la medida revoluciones frente a velocidad. 71
Figura 4-2. Coeficientes de Spearman revoluciones frente a velocidad. 71
Figura 4-3. Gráficas de los polinomios generados para revoluciones frente a velocidad. 72
Figura 4-4. Modelo revoluciones frente a velocidad sobre el fondo. 72
Figura 4-5. Visualización inmersiva del modelo para revoluciones y velocidad sobre el fondo. 73
Figura 4-6. Incertidumbres en la medida revoluciones frente a consumo de combustible. 74
Figura 4-7. Coeficientes de Spearman revoluciones frente a consumo de combustible. 74
Figura 4-8. Representación de de modelos revoluciones / consumo de combustible. 75
Figura 4-9. Modelo revoluciones frente a consumo de combustible. 75
Figura 4-10. Visualización inmersiva del modelo para revoluciones y consumo de combustible 76
Figura 4-11. Incertidumbres en la medida revoluciones frente a presión del aceite de lubricación. 79
Figura 4-12. Coeficientes de Spearman revoluciones a presión del aceite de lubricación. 79
Figura 4-13. Representación de modelos revoluciones / presión del aceite de lubricación. 80
Figura 4-14. Modelo revoluciones frente a presión del aceite de lubricación. 80
Figura 4-15. Visualización inmersiva del modelo revoluciones / presión del aceite. 81
Figura 4-16. Prueba de la aplicación por personal ajeno al trabajo. 82
Figura 4-17. Código QR con enlace a vídeo del manual de empleo. 83
Figura 5-1.Resumen líneas futuras de desarrollo académicas 86
Figura 5-2. Modelo con variable temporal 87

Figura Anexo I- 1. Emisiones del transporte en la UE. 93

Figura Anexo II- 1. Diagrama de capacidades de los órganos de sostenimiento 101

ÍNDICE DE TABLAS

Tabla 2-1. Tecnologías implicadas en el desarrollo del astillero 4.0 16
Tabla 2-2. Equiparación del sensado en los MMPP de Buques clase BAM de la Armada 23
Tabla 2-3. Datos generales de la Pub. ENM 4810 [30] 30
Tabla 2-4. Leyenda de simbología CAT Marine Power Display 30
Tabla 3-1. Análisis OPVL 35
Tabla 3-2. Matriz de gestión de riesgos empleada 39
Tabla 3-3. Variables medidas en este trabajo 40
Tabla 3-4. Secuencia de órdenes a bordo de la lancha. 41
Tabla 3-5. Matriz de decisión con ejemplos para selección de metodología de modelado 3D. 57
Tabla 3-6. Matriz de gestión de riesgos del procesado del escaneo de Leica 59
Tabla 4-1. Incertidumbre de la velocidad sobre el fondo. 70
Tabla 4-2. Resultados generados para revoluciones frente a velocidad. 71
Tabla 4-3. Incertidumbre del consumo de combustible. 73
Tabla 4-4. Resultados generados para revoluciones frente a consumo de combustible. 75
Tabla 4-5. Cálculo de consumos en ejercicios de navegación por aguas restringidas. 77
Tabla 4-6. Cálculo de consumos en ejercicios de aproximaciones de aprovisionamiento. 77
Tabla 4-7. Cálculo de consumos en ejercicios de evoluciones. 78
Tabla 4-8. Incertidumbre de la presión del aceite de lubricación. 78
Tabla 4-9. Resultados generados para revoluciones frente a presión del aceite de lubricación. 79

1 INTRODUCCIÓN Y OBJETIVOS

1.1 Motivación

El concepto de Armada 4.0 es definido por el Excmo. Sr. Almirante (Retirado) Manuel Garat Caramé en la conferencia "Universo 4.0. Impacto en las organizaciones" de la Semana Naval de Madrid 2018 como: la nueva "gestión orgánica de la Armada para producir capacidades, siendo éstas las demandadas tanto por la Flota como por los organismos de apoyo a la misma"[1]. La principal capacidad a la que se refiere es la optimización de la relación interdepartamental de la Armada, integrando una metodología de trabajo usuario-sistema-usuario, con dependencia de una única base de datos, común a todo su espectro [1].

El empleo de bases de datos que recojan el funcionamiento de los equipos y sistemas de las unidades suele estar orientado principalmente a labores de sostenimiento más que a las operativas, aunque pretenden ser una herramienta multidisciplinar. La calidad y propiedad del dato en una comunidad 4.0 como es la Armada es un factor de especial relevancia para el conocimiento del estado de los buques y unidades de las que dispone la Flota. Su procesamiento permite llevar a cabo labores de monitorización, simulación y predicción de su funcionamiento, una capacidad que ha encontrado en el gemelo digital de las futuras fragatas F-110 su propulsor.

Otros proyectos en el ámbito de defensa han representado un punto y aparte en el tratamiento de la información. Este es el caso de los proyectos ATAVIA y SOPRENE, así como las tareas de vigilancia que cada vez se realizan con mayor frecuencia a nivel unidad, centro u organismo (UCO).

La Escuela Naval, por su parte, como casa madre de todos los oficiales de la Armada, debe mostrar la iniciativa en todas las tendencias innovadoras de interés, y la transformación digital es una de sus ramas de trabajo. La ubicación actual del Centro de Inteligencia Artificial de la Armada (CIA2) y la disponibilidad del personal hacen de éste el entorno idóneo para tomar la iniciativa con un proyecto aplicable a un gran abanico de unidades y útil tanto para labores de mantenimiento predictivo como para la formación en esta materia de los oficiales que ocuparán puestos en la orgánica del sostenimiento.

Más en detalle sobre el alcance de esta herramienta, la posibilidad de integrar a las Lanchas de Instrucción en un sistema de control de sus variables de operación es una de las líneas de trabajo que se

[1] La Armada, al igual que el Ejército de Tierra y el Ejército del Aire y el Espacio, se divide en las ramas de Fuerza (Flota en el caso de la Armada), Apoyo a la fuerza y Cuartel General.

explotarán con la llegada de las dos nuevas unidades ya sensorizadas en 2023, por lo que se ha considerado la opción de que las anteriores Rodman también tengan algún tipo de control similar.

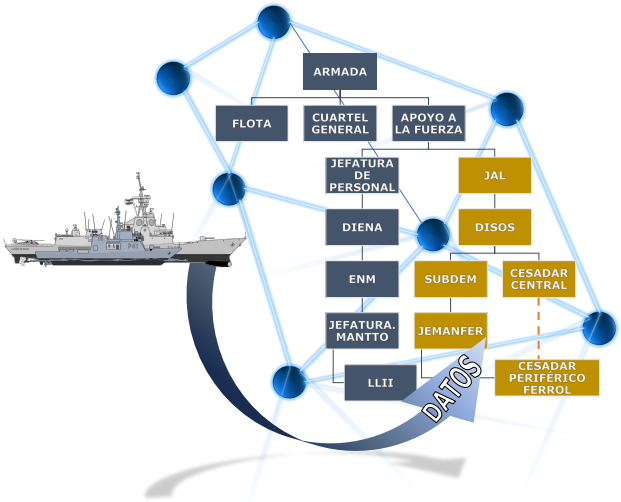


Figura 1-1. Organigrama general de dependencias de los datos de las plataformas de la Armada
Elaboración propia

1.2 Objetivos

El presente trabajo desarrolla una serie de objetivos troncales, o ligados de forma directa al proyecto, así como cumplimenta otras necesidades que, por su compatibilidad y capacidad de integración, han sido consideradas de interés para su desarrollo.

Los objetivos principales se basan en los siguientes:

- Diseño de una base de datos capaz de modelar el comportamiento de los sistemas de los buques en sus diferentes situaciones de operación.
- Desarrollo de una aplicación con capacidad de visualización de los datos e interacción con ellos.

Los objetivos secundarios comprenden:

- Programación de una visualización de un modelo interactuable en 3D para mostrar una visualización inmersiva del funcionamiento.
- Implementación de técnicas de machine learning como método para la generación de datos coherentes y que suplan la necesidad de sensorización de la plataforma en los trabajos de conceptualización.

1.3 Metodología

La aproximación al problema se realiza desde varios campos por su diferente naturaleza que confluye en un único producto final. La descripción de esta metodología, en secuencia cronológica, sería:

- Minería de datos en hojas de cálculo (MS Excel) a partir de registros de buques sensorizados.
- Obtención de modelos de comportamiento del buque en base a la minería previa y que sean extrapolables a las LLII por medio de técnicas de inteligencia artificial con Python.

- Desarrollo de una interfaz de visualización de valores numéricos, que se puedan seleccionar y con los que pueda interactuar, incluyendo una opción de vista a bordo de la lancha para una experiencia más inmersiva e interactuable.
- Para esta última ventana de visualización, es necesario la obtención de un modelado 3D, en este caso realizado por técnicas de ingeniería inversa, y el tratamiento de este modelo, para el que se empleará el software Unreal Engine.

1.4 Estructura de la memoria

La estructura de los contenidos de la memoria parte de la conceptualización del trabajo en este apartado introductorio para desenvolverse en los capítulos posteriores:

- Capítulo 1: Introducción y objetivos. Presente apartado con información general del proyecto.
- Capítulo 2: Estado del Arte. Revisión bibliográfica de los contenidos necesarios para el desarrollo del trabajo.
- Capítulo 3: Desarrollo del TFG. Descripción de los procesos y tareas realizados.
- Capítulo 4 Resultados/Validación/Prueba. Comprobación de funcionalidades a nivel usuario.
- Capítulo 5. Conclusiones y líneas futuras. Opciones de mejora u otros proyectos que tomen este trabajo como punto de partida.

Además de esta división, se contempla la inclusión de Anexos que complementen las bases de lo descrito a lo largo de la memoria, siendo remarcables por su entidad propia los dos primeros:

- Anexo I. Implicaciones Sociales y/o Económicas y/o Ambientales. Desarrollo del impacto del trabajo en las áreas del epígrafe.
- Anexo II. Reflexiones éticas y sociales. Recopilación de las consideraciones en el plano ético y social surgidas a lo largo del trabajo. Caso específico de la implantación de este software como herramienta de apoyo en la Escuela Naval Militar y el significado del dato desde un punto de vista social.

2 ESTADO DEL ARTE

2.1 Contexto en la Armada

La Armada es una de las tres organizaciones que componen la estructura orgánica de las Fuerzas Armadas, responsable de aportar las capacidades específicas del ámbito marítimo a la estructura operativa, dependiente del Estado Mayor de la Defensa (EMAD). [2]

Esta institución se encuentra bajo el mando del Almirante General (OF-9)[2] Jefe del Estado Mayor (AJEMA), y encuadra a sus componentes bajo los diferentes mandos de Almirantes subordinados. La Jefatura de Apoyo Logístico (JAL) es una de estas dependencias, responsable de la gestión y control de los recursos materiales de la Armada durante la totalidad del ciclo de vida (CdV) de los sistemas.

La función logística es un arte más de la guerra de una elevada complejidad por su naturaleza previsora y facilitadora para las fuerzas operacionales. Su origen se remonta al interés las tropas napoleónicas por asegurar la sostenibilidad de las unidades del frente, llegando a entenderlo como una necesidad básica del combatiente [3]:

"The amateurs discuss tactics; the professionals discuss logistics" (Napoleon Bonaparte).

Sobre la composición de la JAL, ésta se descompone en las jefaturas de los arsenales de apoyo a la Flota, Ferrol, Las Palmas, Cádiz y Cartagena y las direcciones de sostenimiento (DISOS), infraestructura (DIN), ingeniería y construcciones (DIC) y gestión económica (DIGEC). La DISOS es la responsable de asegurar el aprovisionamiento y mantenimiento de las unidades, funciones que han evolucionado con las nuevas tecnologías hacia la monitorización de actividades en tiempo real.

En el "Concepto de Apoyo Logístico" de 2017 establece los principios tecnológicos que deberán ser fuente de inspiración de los Procesos de Trabajo Logísticos (PTL). Estos son:

- Interoperabilidad: se procurará la interconexión de elementos sobre una plataforma digital.
- Virtualización: deberá procurarse la integración de los PTL en modelos de gemelos digitales.
- Automatización: reducción del planeamiento y control de los PTL mediante algoritmos y sistemas inteligentes.
- Capacidad de trabajo en tiempo real: transmisión de datos en tiempo real para asegurar la disponibilidad de los datos.
- Fiabilidad: elección de un sistema óptimo que combine los mantenimientos predictivos, preventivos y correctivos.

[2] Código NATO/OTAN

- Modularidad: los sistemas deben ser independientes para realizar PTL selectivos sin afectar a terceros sistemas.
- Orientación de los servicios al cliente: los PTL deberán conjugar una reducción de costes con la confianza en proveedores y contratistas. [4]



Figura 2-1. Visión de la JAL 4.0
Elaboración propia

Los gemelos digitales u otros sistemas de visualización de datos de operación de buques en servicio son proyectos de interés para la DISOS, mientras que, para las futuras unidades, la responsabilidad de su implementación desde la fase de definición de necesidades del barco recae en la DIC.

En el caso de la Armada, la digitalización y monitorización del dato es un área de trabajo ya existente, fundamentada a bordo de los propios buques en el Sistema Integrado de Control de Plataforma (SICP) y, de forma remota, por el Centro de Supervisión y Análisis de Datos de la Armada (CESADAR) en unidades de apoyo en tierra.

2.2 Sistema Integrado de Control de Plataforma

Se trata de un sistema que permite controlar y monitorizar todos aquellos sistemas del buque que no formen parte del sistema de combate: propulsión, planta eléctrica, auxiliares y control de averías. En la incorporación del SICP, las consolas de a bordo presentan una interfaz gráfica entre el operador y los autómatas programables cuyas prestaciones han sido objeto de una evolución progresiva desde su incorporación en el primer buque de la Armada con esta capacidad, el Buque de Aprovisionamiento en Combate (BAC) Patiño.

El SICP cuenta con puestos de operador fijos distribuidos en las diferentes secciones del buque, y presenta la posibilidad de establecer una conexión al sistema mediante un PC externo, en sus entradas *PC-IN*. En cuanto al hardware disponible, el sistema cuenta con consolas CONAM con 1 y 2 pantallas y equipos de almacenamiento de datos que se encargan de registrar los históricos del buque.

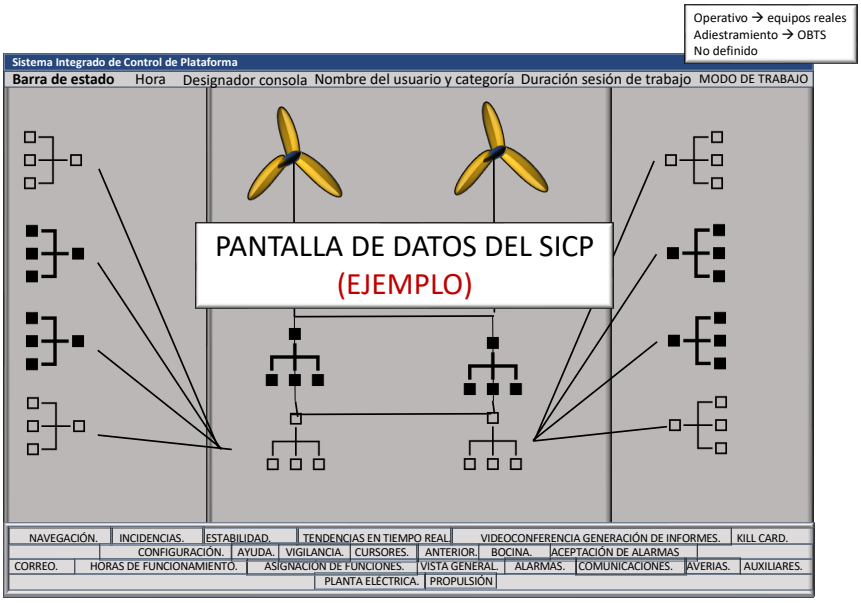


Figura 2-2. Esquema de una ventana de visualización del SICP
Elaboración propia

El sistema tiene asignados grupos de usuarios para dar acceso a las diferentes tareas que se pueden realizar. El de mayor nivel es el de supervisor y super-usuario y el de mayores restricciones, el nivel usuario, empleado para generación de informes, gestión de alarmas, históricos de funcionamiento y tendencias de los equipos [5].

2.3 Centro de Supervisión y Análisis de Datos de la Armada

El CESADAR, con dependencia funcional de la DISOS, se compone de un total de cuatro centros periféricos y una unidad central, ubicada en Cartagena y con dependencia orgánica de la jefatura de mantenimiento de su Arsenal. Su trabajo se fundamenta en la obtención de los datos registrados por buques, definiendo los procedimientos para su recogida, organización, mantenimiento y análisis. Estos datos son registrados a bordo por el Sistema Integrado de Control de Plataforma (SICP) y por el Sistema de Mantenimiento Basado en la Condición (SMBC), para luego ser transmitidos vía satélite al CESADAR Central, que a su vez será el responsable de compartir esas bases de datos con los CESADAR periféricos, basados en Cartagena, Ferrol, Cádiz y Las Palmas.

El funcionamiento de esta unidad está regulado en la Instrucción Permanente 001/2011, de 15 de marzo, del Almirante Jefe de Apoyo Logístico sobre organización y funcionamiento del sistema CESADAR. Como fin último, este centro debe servir como una herramienta para la toma de decisiones en las labores de mantenimiento, para lo que se han desarrollado proyectos de gestión de las bases de datos como las aplicaciones ATAVIA y SOPRENE. [6]

2.3.1 Automatización de Tareas de Vigilancia y Análisis (ATAVIA)

Es una aplicación, desarrollada por el CESADAR que tiene como objetivo el analizar todos los sistemas y equipos que montan los buques y están sensorizados. Es accesible a través de la Infraestructura Integral de Información para la Defensa (I3D) como parte del Sistema Logístico Integrado (SLI).

ATAVIA almacena, procesa y analiza los datos provenientes de los equipos y sistemas sensorizados a bordo de los buques, accesibles por el usuario en función de su perfil de acceso. Estos datos se visualizan en un rango temporal determinado, y algunos ejemplos son: probabilidad de ocurrencia de modo de fallo, criticidad, recomendación de solución/actuación y avisos de anomalías de

funcionamiento, estadísticas de navegación y de los análisis de las materias de servicio, así como cuadros de mando a sus usuarios.

El usuario final (como ente corporativo) son las unidades de mantenimiento (jefaturas dentro de arsenales u otras unidades de apoyo) y la información que recibe es tratada mediante técnicas de BI (Business Intelligence), que sirven de ayuda a la toma de decisiones de mantenimiento.

Dada la naturaleza del programa, como una aplicación on-line, la necesidad de un ancho de banda suficiente para la transmisión de datos es un reto al que se enfrentan las unidades, un problema al que un sub-programa de ATAVIA puso solución, el Módulo Embarcado de Vigilancia de Mantenimiento (MEVIMAN). El MEVIMAN cumple con las tareas de ATAVIA, con la diferencia de ser operado desde el propio buque. [7]

2.3.2 Sostenimiento Predictivo basado en Redes Neuronales (SOPRENE)

La Dirección General de Armamento y Material (DGAM) establece como objetivo del programa SOPRENE el proporcionar un sistema capaz de, por medio de técnicas de inteligencia artificial, mejorar la predicción de eventos asociados a fallos o el malfuncionamiento de los equipos embarcados, así como presentar su operatividad. Este objetivo es desglosado por la multinacional y colaboradora en el proyecto INDRA Sistemas S.A. en cuatro:

- Conocimiento: la emisión de datos al CESADAR para su tratamiento.
- Control: desarrollo por parte de INDRA de un demostrador de plataforma de código abierto formando al personal del CESADAR en su uso
- Análisis: decisión de las técnicas a emplear sobre los datos obtenidos.
- Demostración: dar a conocer la operatividad de equipos u otras variables de forma amigable para el usuario no especializado de los buques o ajenos al CESADAR.

Sobre su evolución y disponibilidad, esta herramienta es aplicable únicamente a las fragatas de la clase “Álvaro de Bazán” y Buques de Acción Marítima de la clase “Meteoro” desde el año 2019, momento a partir del cual los esfuerzos pasaron de estar centrados en la creación de la plataforma a la analítica de datos. [8]

2.4 La industria 4.0

Con la llegada de la máquina de vapor de manos de James Watt y Thomas Newcomen en 1760, su introducción en la producción da pie al mecanizado de tareas que previamente requerían, por lo general, del esfuerzo humano o animal. Sumando este invento al descubrimiento de materiales metálicos y combustibles (hierro y carbón), entre los siglos XVII y XIX tiene lugar la conocida como Primera Revolución Industrial.

Dado el incremento en la producción a lo largo de las siguientes décadas, la solución para implementar procesos con menores costes y tiempos de procesado vino de mano de Henry Ford con las cadenas de montaje. Esta idea se basaba en la división de tareas y la producción en masa, incorporaciones a la industria que llegaron de forma simultánea a la explotación de un mayor número de recursos energéticos, como la electricidad y el petróleo, y otros avances de carácter tecnológico, como la radio y el teléfono. Es este cúmulo de avances el que marca un hito, siendo conocido este periodo de mediados del siglo XIX como la Segunda Revolución Industrial.

El avance tecnológico del siglo XX permitió el continuo desarrollo de la industria a un ritmo superior al que se venía observando en las décadas anteriores. La explotación de la informática, la electrónica y las comunicaciones, conocidas como Tecnologías de la Información y de la Comunicación (TICs), permitió que se produjese una Tercera Revolución Industrial en la que tendrían un papel de especial relevancia la globalización, la automatización de procesos y la aparición de internet. Cabe destacar en

un trabajo de este contexto el origen militar de la red, fruto del proyecto ARPANET del Departamento de Defensa (DoD) de los Estados Unidos en 1969, hasta su escisión en MILNET y NSFNET, y la evolución de la última de las dos en el actual internet.

Tras la llegada de la crisis económica de 2007/2008, la actividad industrial se vio fuertemente condicionada y reducida, hasta que por iniciativa del gobierno alemán se intentó buscar una solución tecnológica al problema. Durante la Feria de Hannover de 2011 se acuña el término industria 4.0, que poco después se verá extendido por otros países gracias a la labor de la globalización. [9]

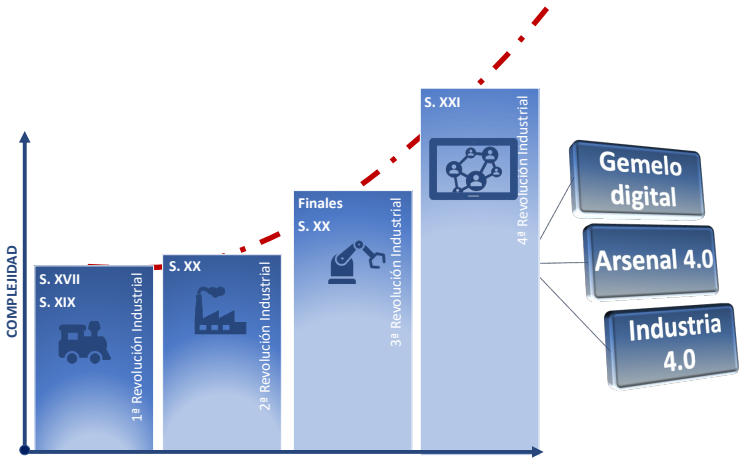


Figura 2-3. Escala de tiempos de las diferentes revoluciones industriales
Elaboración propia

Por su parte, la industria española del naval ha tomado también la iniciativa al adaptarse a esta última revolución. La empresa Navantia hace público en su Plan Estratégico Nacional 2018-2022 su interés por el desarrollo tecnológico del sector. El que en un mayor grado concierne a la actualización de la industria es el segundo pilar de los tres en los que se fundamenta esta estrategia, siendo estos la potenciación de la cartera de Navantia, la eficiencia operativa y transformación digital y el rejuvenecimiento y nuevo marco laboral. [10]

Sobre esta transformación digital, es aquí donde Navantia hace hincapié en el concepto del Astillero 4.0 asignando los siguientes conceptos:

- Fábrica inteligente y cadena de suministro 4.0: implementación de inteligencia artificial y técnicas analíticas para obtener una gestión precisa del seguimiento de programas, desde niveles administrativos hasta la seguridad en tareas del operario.
- Productos y servicios inteligentes: giro entorno al buque inteligente, sensorizado y con aplicaciones de mantenimiento predictivo/preventivo y seguimiento, todo combinado con la máxima automatización en operación.
- Personas y cultura 4.0: renovación/actualización de competencias en los perfiles profesionales de los recursos humanos.
- Procesos corporativos 4.0: desde actividades productivas e industrializadas hasta aquellas administrativas como empresa.
- Plataformas y arquitectura digital, empleo de nuevas tecnologías y técnicas como la sensorización o inclusión de inteligencia artificial en simulaciones propias de los trabajos realizados.



Figura 2-4.Concepto de astillero inteligente con fragata en grada en Ferrol
Elaboración propia

A modo resumen, también contenidas en el marco de este Plan Estratégico, las tecnologías de las que dependerá el astillero 4.0 son:

Tecnología	Aplicabilidad	Tecnología	Aplicabilidad
Big Data	Predicción de comportamientos en base a almacenamiento masivo de datos	Vehículos autónomos	Optimización de procesos de transporte
Internet de las Cosas (IoT)	Productos y herramientas con conectividad	Realidad Virtual & Aumentada	Herramientas de visualización inmersiva
Nube	Almacenamiento común del dato	Modelización y simulación	Predicción y análisis de comportamientos
Impresión 3D	Producción mediante técnicas aditivas in situ	Inteligencia Artificial	Aprendizaje automático
Nuevos materiales	Mejores propiedades	Ciberseguridad	Protección
Plataforma digital	Conectividad desde un soporte informático común	Robótica	Automatización industrial
		Blockchain	Seguridad económica en la red

Tabla 2-1. Tecnologías implicadas en el desarrollo del astillero 4.0
(elaboración propia a partir del Plan Estratégico de Navantia 2018-2022)

2.5 El gemelo digital en la Armada

La aproximación al gemelo digital (GD o DT por sus siglas en inglés) es la misma que un enlace entre el mundo físico y un entorno virtual, con capacidad de presentación y monitorización de los sistemas de información que han sido sensorizados para su posterior análisis.

Esta herramienta tiene su origen en el programa Apollo, de la National Aeronautics and Space Agency (NASA) en los años 70 del siglo XX con el fin de replicar el funcionamiento de las tecnologías aeroespaciales desde la Tierra. Sin embargo, no es hasta el año 2003 cuando se toma la iniciativa para integrar los gemelos digitales en tareas del ámbito civil. De la mano del Dr. Michael Grieves con su presentación del proyecto *Product Lifecycle Management* (PLM), se pretende obtener una representación dinámica de los sistemas durante sus cuatro fases del ciclo de vida; diseño, producción, operación y retirada. En el año 2012, la NASA incluye los gemelos digitales en su programa de mantenimiento de aeronaves, con técnicas de simulación. [11]

Es a partir del año 2015 cuando el interés por la explotación de sistemas dinámicos de monitorización presenta un gran impulso, siendo actualmente una de las 10 tendencias tecnológicas de mayor valor estratégico de acuerdo con el informe de la consultora tecnológica Gartner para el año 2023. Este listado incluye los gemelos digitales en su puesto número 9 bajo el subgrupo del metaverso, aunque el resto de las tecnologías están también ligadas al desarrollo de este tipo de bases de datos.

En el caso particular de la Armada, actualmente se contempla el gemelo digital como "parte intrínseca del programa de la fragata F110", abriendo las posibilidades de empleo de realidad virtual o aumentada y de asistencia remota, así como integrar la impresión 3D. Las mencionadas son algunas de las aplicaciones de este desarrollo con impacto directo en las unidades de la Flota. Si bien es cierto, existen otras implicaciones extendidas a un ámbito más general en la estructura del apoyo a la fuerza. La de mayor impacto es la implantación del modelo del Arsenal 4.0, definido recientemente por la JAL en la "Visión del AJAL del Arsenal inteligente" del 26 de Enero de 2023. [12]

Este documento habla del proceso de actualización como un modelo general "evolutivo" asentado en los siguientes pilares:

- Hibridación (H): no se pretende llevar a cabo una completa suplantación de las técnicas actuales por otras de mayor desarrollo tecnológico, sino que se busca la convivencia con los sistemas físicos.
- Avance de una organización colaborativa a otra inteligente (I): contando con capacidad de mantener el control de la cadena de valor que suponen las acciones y procesos que desarrolla el propio arsenal.
- Eficacia operativa y eficiencia en gestión en tres vertientes (3E), que deben procurarse de forma balanceada sin sacrificio de ninguna de ellas:
    - Eficacia Operativa: las tareas de sostenimiento garantizarán la operatividad de las unidades con resultados de mayor calidad y menor coste de recurso temporal.
    - Eficiencia Económica: deberá ser un sistema capaz de maximizar el rendimiento de la disponibilidad del presupuesto existente para las unidades y sostenimiento del propio arsenal.
    - Eficiencia Medioambiental: los factores de mayor relevancia en este ámbito son la mínima polución y el objetivo de mínimo gasto energético. Se busca superar los estándares más exigentes aplicables en pos de una óptima gestión de residuos y mayor eficiencia energética, llegando incluso a valorar la autogeneración de energía no contaminante.
- Gestión automatizada por procesos. Optimización de tareas cíclicas o recurrentes para mejorar la eficiencia y eficacia del arsenal.
- El recurso humano como centro de gravedad. La formación del personal es esencial para garantiza que la tecnología actúe como herramienta facilitadora para el ser humano, que seguirá siendo el centro del apoyo logístico.
- La convivencia de tecnologías maduras con otras menos desarrolladas. Teniendo en cuenta la existencia en el Listado Oficial de Buques de la Armada (LOBA) de unidades entregadas en el marco temporal del último siglo, con las diferencias tecnológicas entre sí que ello implica.

- Conocimiento en tiempo real de la situación logística de las unidades y capacidad de predicción. Ventaja y anticipación con programación de acciones correctivas acordes a las necesidades de la unidad.

2.6 Modelos de operación de Buques, el MMG

La simulación numérica de los movimientos dinámicos de los buques se realiza en base a diversos modelos desarrollados con herramientas de computación, existiendo modelos más acertados para casos concretos. En el caso de la maniobrabilidad, un modelo altamente empleado es el Manoeuvring Mathematical Modeling Group, MMG. [13]

En el caso de los análisis de dinámica de fluidos computacional (CFD[3] por sus siglas en inglés) en buques, por lo general se tiende a emplear únicamente las componentes de velocidad y aceleración del flotador, ya que la contribución de fuerzas dinámicas dificulta la definición de un modelo concreto. De esta forma, se tiende a trabajar con procesos cuasiestáticos, definidos en casos como las ecuaciones del movimiento con 3 grados de libertad referida al centro de gravedad del buque (1).

$$\begin{cases} m(\dot{u}_G - v_G r) = X_G = X_{GA} + X_{GS} \\ m(\dot{v}_G + u_G r) = Y_G = Y_{GA} + Y_{GS} \\ I_{zz}\dot{r} = N_G = N_{GA} + N_{GS} \end{cases} \quad (1)$$

Siendo m la masa del flotador; I_{zz} el momento de inercia en el eje de guiñada; u_G , v_G y r las componentes de velocidad del cuerpo; $\dot{u}_G$, $\dot{v}_G$ y $\dot{r}$ las de aceleración; X_G, Y_G y N_G son las fuerzas hidrodinámicas, modificadas por el sufijo "A" en el caso de las debidas a la propia aceleración, y por el sufijo "S" en el caso de las debidas a la velocidad.

El modelo MMG fue desarrollado entre 1976 y 1980 por el grupo japonés de investigación homónimo, relacionando las características del casco, planta propulsora y timón.

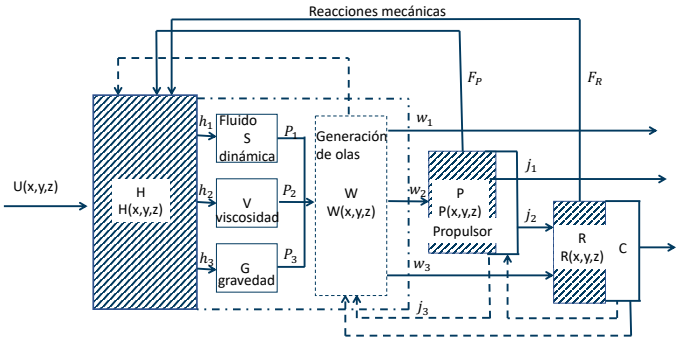


Figura 2-5. Bloques del modelo MMG
Elaboración propia en base a **[13]**

Las ecuaciones por las que se rige el MMG se ajustan generalmente a los modelos de regresión polinómica, como en el caso de la ecuación del coeficiente de empuje del propulsor (**2**), que es una ecuación de segundo grado con el avance del propulsor J como variable independiente.

$$K_T = a_0 + a_1 \cdot J + a_2 \cdot J^2 \quad (2)$$

[3] Los análisis CFD permiten, por medio de matemáticas aplicadas, el estudio de la interacción entre fluidos (líquidos o gaseosos) y el cuerpo de estudio.

2.7 Software empleado en gemelos digitales

En el mundo de la industria, no existe hoy en día un consenso sobre las funcionalidades y alcance que deben lograr los gemelos digitales. Es por ello, que las metodologías y tecnologías que son de aplicación en este campo son variadas en función del interés del producto final. Sí que existe, sin embargo, una idea general surgida en los últimos años y que dicta los siguientes mínimos para los gemelos digitales:

- Definición y simulación de trayectorias, considerando el entorno y el propio sistema.
- Simulación y control de la estación o sistema.
- Definir y simular la automatización de la estación.
- Definir y simular las capacidades de comunicación y procesamiento de datos para su intercambio [14].

Todos estos son factores son variados y no siempre existe un software capaz de satisfacer todos estos requisitos. Se considera por tanto que los gemelos digitales precisan de un desarrollo multidisciplinar y requieren de un conocimiento exhaustivo en las diferentes materias asociadas al CdV del proyecto, en especial a su operación.

2.7.1 Visualización en 3D

En los casos en los que se trabaja con una visualización tridimensional del sistema, replicando su funcionamiento, existen programas CAE (Computer Aided Engineering) que facilitan la integración de esta simulación con el movimiento del sistema y sus datos en cada momento. Por lo general, es software especializado de empresas como las conocidas Autodesk Inc. o Siemens Inc., pero otros desarrolladores han optado por entrar en este campo en auge, como desarrolladores de simuladores y videojuegos y creadores de software en general, como Microsoft ® con su solución MS Azure digital twins. Se aprecian por tanto dos posibilidades:

- Empleo de aplicaciones específicas de ingeniería all-in-one: su principal inconveniente es el coste del software y la falta de libertad en el desarrollo por no ser herramientas open-source. Los resultados visibles están predefinidos y la interfaz del usuario no suele ser personalizable, una opción valorada cuando se trabaja en un proyecto bajo demanda, satisfaciendo las necesidades del consumidor.

- Empleo de software alternativo como complemento: las opciones que ofrecen son más variadas ya que el entorno de trabajo es libre. Implican un mayor esfuerzo por la integración de los sistemas ya que se acostumbra a trabajar desde cero por la novedad de estos procedimientos. Las interfaces son totalmente personalizables y el alcance de los proyectos que pueden manejar está limitado por la imaginación y requisitos del cliente. Algunos ejemplos de software empleado en este rango son los programas de desarrollo de videojuegos y simuladores, como Unity 3D y Unreal Engine, de la empresa Epic Games®. Este último además ha desarrollado en los últimos años la herramienta "Twinmotion", un puente entre Unreal Engine y programas CAD, permitiendo este factor de visualización de los gemelos digitales. [15]

2.7.2 Simulación

La principal tecnología que ha permitido la implantación de los gemelos digitales es el internet de las cosas (IoT), que en el ámbito industrial se denomina *Industrial Internet of Things* (IIoT), y que ha hecho posible la conexión de los entornos físicos con el mundo virtual. De esta forma, la computación para procesos de simulación queda relegada al plano de las predicciones en vez de la operación habitual. En otros casos, los sistemas no están siendo monitorizados mediante sensores *on-line*, por lo que es necesario simular su comportamiento. Ésta última es la opción elegida por la mayoría de los programas CAE que integran gemelos digitales hoy en día, trabajando sobre sus bases de datos y funciones

matemáticas genéricas. En el caso de los desarrollos con visualizaciones desde software independientes, se puede trabajar con estas leyes genéricas, introducidas por el propio desarrollador, o se pueden integrar modelos de operación en base a las observaciones de los buques.

2.8 Inteligencia Artificial

A pesar de no existir una definición homogénea para esta tecnología, Indra la define como el área de la ciencia encargada de dotar de inteligencia a las máquinas con capacidad de computación para así potenciar el esfuerzo humano. [16] Su aplicabilidad a gran número de estudios y tareas es una de las principales razones por las que no existe consenso sobre su descripción, aunque para trabajos del índole de la simulación y monitorización, los de mayor interés son el aprendizaje de las máquinas (conocido como machine learning) y la lógica difusa (fuzzy logic).

La lógica difusa es un método que permite tratar de forma científica conceptos cualitativos, mediante reglas de tipo antecedente-consecuente. [17]

El Machine learning es una técnica basada en el empleo de estadística, teoría de probabilidad, optimización u otros métodos matemáticos con el fin de alcanzar inferencias en base a una serie de datos proporcionados. Por su relación con el área de los gemelos digitales y modelado es objeto de mayor detalle en el subapartado Machine Learning. [18]

2.9 Machine Learning (ML)

Profundizando en esta rama de la inteligencia artificial, su empleo acostumbra a ser apropiado en casos como los siguientes:

- Si no existen conocimientos explícitos en la materia para la obtención de un algoritmo que resuelva el problema, pero si una cantidad suficiente de ejemplos de la tarea a estudiar.
- Si el problema a resolver varía con el tiempo.
- Si el flujo de datos es continuo y contiene información renovada apta para validar la fiabilidad del modelo.

De la misma forma que en el mundo físico, el aprendizaje debe partir de una base, la adquisición del conocimiento en base a la experiencia, lo que define tres clases de machine learning: el aprendizaje supervisado, el no supervisado y el aprendizaje por refuerzo.

- El aprendizaje supervisado consiste en el empleo de proceso iterativo de aprendizaje, siendo guiado por un conjunto de datos que previamente han sido etiquetados y clasificados, los que serán los datos de entrenamiento. Se entiende como etiqueta la asociación a los datos de entrada de una solución al problema válida. A partir de este resultado conocido, el algoritmo aprende comparando su output o salida con el valor deseado, y realiza los ajustes necesarios para cada error.
- El aprendizaje no supervisado se diferencia del anterior en que los datos empleados en el aprendizaje no han sido previamente clasificados, valiéndose únicamente de ellos para analizar el conjunto y establecer relaciones entre ellos. Este modelo está más orientado a tareas de clasificación, mediante algoritmos como el agrupamiento (clustering), que particiona los grupos de datos que contienen características similares.
- El aprendizaje por refuerzo emplea métodos de acción-recompensa almacenando los patrones de éxito y adaptándose a ellos, de forma que la tasa de aciertos vaya creciendo hasta obtener un modelo con una fiabilidad aceptable.

En el mundo del ML, se distinguen técnicas de clasificación y de regresión. Las primeras permiten la diferenciación de datos entre tipos que pueden asociarse a un conjunto discreto, por ejemplo, la predicción del tipo de buque con datos AIS, como se desarrolla en el trabajo de fin de grado de la fuente [19]. En el caso de la regresión, esta permite la predicción de valores numéricos asociados al

comportamiento del entorno de estudio, el caso de las variables de operación de un buque, como se explica en el apartado Técnicas de ML aplicadas a la regresión.

2.9.1 Fases del proceso de aprendizaje

Para la obtención de un modelo lo suficientemente fiable para operar conforme a los requisitos del trabajo, la metodología deberá seguir unos pasos concretos propios del machine learning. Estos son:

- Obtención del dato.
- Limpieza y preprocesado de la información a analizar.
- Entrenamiento del modelo.
- Validación del modelo.
- Mantenimiento y mejoras.

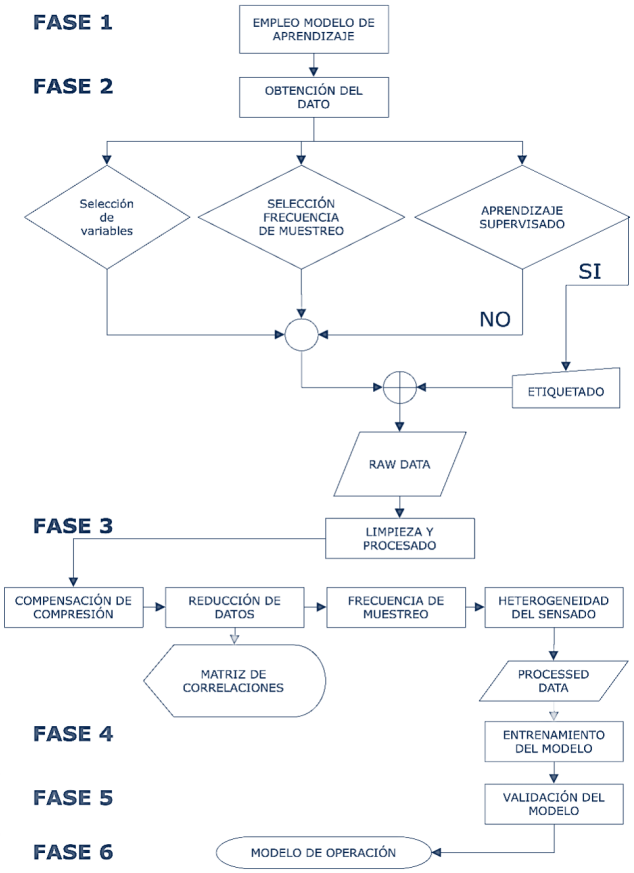


Figura 2-6. Diagrama de flujo del proceso de aprendizaje
Elaboración propia. Simbología ANSI

2.9.1.1 Obtención del dato

La obtención de datos consiste en recabar los registros que servirán como referencia de aprendizaje al sistema. Las decisiones que tomar en esta fase son la elección de variables medibles, la frecuencia del muestreo y, en el caso de sistemas de aprendizaje supervisado, el etiquetado. Esta última tarea puede requerir de personal experto y mayores costes. Como caso práctico se tomará el programa SOPRENE, explicado en el subapartado Sostenimiento Predictivo basado en Redes Neuronales (SOPRENE) En esta fase, la toma de datos se realiza directamente en los sensores de las fragatas F100, que son transmitidos a las dependencias del CESADAR para su procesado.

2.9.1.2 Limpieza y preprocesado de la información a analizar

La limpieza y adaptación de los datos trata de adecuarlos al formato que admitirá el modelo, tanto en escala como en el tipo de variables (cuantitativas, nominales, etc.). El tratamiento de los datos debe procurar separar del proceso tanto los valores incorrectos (fuentes de ruido o errores de medida) como aquellos datos que no aporten valor, por ejemplo, valores constantes de menor relevancia. Se puede dar el caso también de que en un set inicial de variables exista alguna pérdida de datos en alguna de las mediciones (missing data). En este caso, será necesario eliminar estas muestras o derivar sus valores a partir de la información del conjunto completo de datos, o bien de variables similares especialmente señaladas. Sobre el trabajo realizado en SOPRENE, éste se enfrenta a cuatro tratamientos:

- La compensación de compresión: en las minerías de datos iniciales del programa, se advirtió la falta de variables en ciertos campos. Esto se debía a que la propia base de datos a la que se transfería la información obviaba el valor de los datos que hubiesen permanecido invariables desde el anterior registro. El problema fue solucionado sustituyendo el dato vacío por el anterior registrado en el data set que es importado a la siguiente fase.
- Reducción de datos: se han elegido para el modelo de SOPRENE los datos que guardan altas correlaciones entre sí, es decir, presentan dependencias directa o inversamente proporcionales. Un ejemplo de metodología para el cálculo de concordancia entre variables son los coeficientes de correlación de rangos de Spearman definido en la ecuación (3). El primer proceso determinando esta analogía entre valores consiste en escalarlos de forma que sean comparables. Esto se lleva a cabo asignando una orden con números enteros para valores únicos en el conjunto. Para datos repetidos en el mismo set, se utilizan números fraccionarios mixtos calculados como el valor de la posición que correspondería a un dato único sumado la fracción unitaria de denominador la cantidad de valores repetidos.

$$r_s = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)} \qquad (3)$$

Siendo r_s el coeficiente a obtener, d_i cada una de las diferencias entre la posición jerarquizada para cada variable con su etiqueta equivalente y n el número de datos a relacionar. A partir de estos valores es posible crear una matriz de coeficientes que sirva de ayuda a la selección de variables implicadas en el modelo.

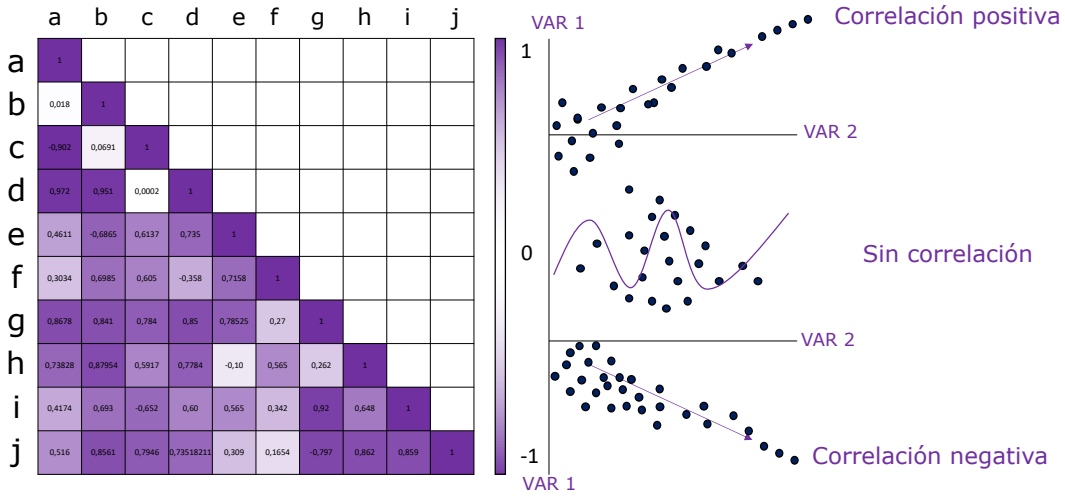


Figura 2-7. Coeficientes de correlación y tipos de correlación entre 2 variables VAR 1 frente VAR 2.
Elaboración propia

En el caso del programa de sostenimiento del CESADAR, el manual de usuario de SOPRENE realiza sus cálculos respecto a los datos de la fragata F-101 "Álvaro de Bazán". Como ejemplo, de la matriz de correlación del generador 1A del buque, en este documento se observan valores de r_s superiores al 90%, como la potencia (reactiva, de r_s=0,96 y activa, de r_s=0,95) y la intensidad del propio generador. [16] Estos resultados hacen visible la proporcionalidad entre variables que definen las ecuaciones empleadas para su cálculo, como se puede ver en las ecuaciones (4) y (5) para obtener potencias activa y reactiva respectivamente [20].

$$P_a = U_{ef} \cdot I_{ef} \cdot \cos\varphi \quad donde\ P_a \propto I_{ef} \qquad (4)$$

$$Q = U_{ef} \cdot I_{ef} \cdot \sin\varphi \quad donde\ P_a \propto I_{ef} \qquad (5)$$

- Frecuencia de muestreo: a partir de los datos preseleccionados en la correlación valores, se tiene en cuenta también la disponibilidad de la información en base a la frecuencia de las mediciones de los sensores, determinando en qué variables existe un mayor número de discontinuidades.
- Heterogeneidad en la disposición y cantidad de sensores: no siempre se dispone del mismo número de sensores entre buques de una misma clase, incluso existen diferencias entre los dos motores principales de una misma unidad. Se pretende evaluar modelos válidos para un espectro lo más amplio posible dentro de la misma clase, por lo que se buscarán variables medibles en diferentes buques.

Unidad	Nº de sensores
BAM42 N1	99 sensores
BAM42 N2	102 sensores
BAM43 N1	104 sensores
BAM43 N2	103 sensores
BAM44 N1	101 sensores

Tabla 2-2. Equiparación del sensado en los MMPP de Buques clase BAM de la Armada
Elaboración propia con datos de **[16]**

El último principio necesario para una correcta adecuación de los datos es el de optimalidad, buscando la eficiencia de estos datos sin llegar a tratar información redundante. La balanza recae en la calidad, y no la cantidad, reduciendo así los costes asociados al procesamiento de datos con resultados igualmente satisfactorios.

2.9.1.3 Entrenamiento del modelo

La fase del entrenamiento ajustará los algoritmos escogidos para representar el modelo a partir de los datos seleccionados para tal fin. El ajuste, por lo general, se realiza por medio de procesos iterativos, cuya duración es función de la complejidad de los algoritmos y de la cantidad de variables consideradas.

La elección del modelo es función de la necesidad que se genera. En el caso del comportamiento de un sistema físico, en base a las variables de operación mecánicas que genera se ajustará a los principios de la regresión, por lo que serán los modelos de estudio en este trabajo.

2.9.1.4 Validación del modelo

Por último, en el ciclo de desarrollo, se encuentra la validación del modelo, en la que se verificará si el resultado alcanza el estándar de calidad necesario. Estos resultados podrán ser correctos, sin necesitar mayor mejora; corresponder a un modelo subajustado, caso en el que precisará de un estudio y aprendizaje más exhaustivos; o ser modelos sobreajustados, en los que, por buscar un error mínimo, el modelo deja de corresponderse con el comportamiento real de las variables.

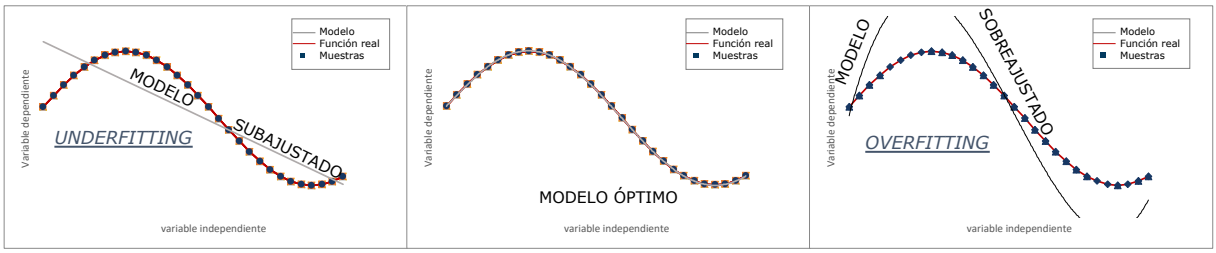


Figura 2-8. Ejemplos de resultados en la validación de un modelo
Elaboración propia

2.9.2 Técnicas de ML aplicadas a la regresión

Dado que el mayor peso de este trabajo recae en el análisis de datos numéricos de conjuntos continuos en mayor medida que en la clasificación de estos, se considera de interés una profundización en los mecanismos de ML que permiten la obtención de estas predicciones, además de considerar técnicas válidas tanto para regresión como clasificación.

2.9.2.1 Regresión lineal y no lineal

La regresión es una técnica que se emplea tanto para previsión como para predicción de eventos, así como para demostración de las correlaciones que existen entre variables dependientes e independientes. Los modelos de regresión se rigen por estos principios para estimar los valores de una variable en función de otra, y se dividen en modelos de regresión lineal simples o de regresión múltiple [21]:

- La regresión lineal simple consiste en la relación de una única variable independiente para predecir el comportamiento de otras. Con ella se distingue la influencia de las variables independientes "x" de la interacción de las variables dependientes "y".

$$g(x) = y = \beta_0 + \beta_1 x + \varepsilon \qquad (6)$$

- Regresión lineal multivariante (MLR): se trata de una técnica estadística que predice el resultado de respuesta ante una variable. Su fórmula es del tipo de la ecuación (7).

$$y = \alpha_0 + \alpha_1 X_1 + \alpha_2 X_2 + \alpha_3 X_3 + \ldots\ldots\ldots\ldots + \alpha_p X_p \qquad (7)$$

  En este tipo de modelo, los errores se distribuyen de forma normal, con una varianza constante y mantiene una relación lineal entre la variable objetivo y la función que lo define. [22]
- Regresión polinómica: este tipo de regresión analiza los polinomios de grado *n* relacionando así variables dependientes e independientes. Es un caso especial de regresión lineal multivariante en la que la ecuación polinómica asociada a los datos se fusiona con las curvas que esbozan los clústers de datos.

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots\ldots\ldots\ldots + \beta_h x^h \qquad (8)$$

2.9.2.2 Error Cuadrático Medio (MSE)

El error cuadrático medio es un estimador que mide los cuadrados de las distancias entre un conjunto de datos y otro deseado. Se trata de un algoritmo de validación de modelos que tiene en cuenta tanto la varianza como el sesgo del estimador, y se rige por la ecuación (9).

$$ECM = h(y) = \frac{1}{n}\sum_{i=1}^{n}(y_{pred} - y_i)^2 \quad (9)$$

En el caso de los algoritmos de regresión lineal, por lo general, el resultado se obtiene mediante la composición de las ecuaciones (6) y (9) de la forma $(h \circ g)(x)$ minimizando el valor del resultado (MSE) con técnicas de obtención de máximos y mínimos, con derivadas y estudio de puntos críticos.

2.9.2.3 Support Vector Machines

Se trata de una técnica de aprendizaje supervisado con algoritmos que permiten el análisis de datos que son empleados para tareas tanto de regresión como clasificación. Gracias a la herramienta del "kernel", puede trabajar en clasificación lineal y no lineal, estableciendo márgenes entre las clases existentes de datos. Estas divisiones son repartidas en espacios multidimensionales que minimizan los errores de clasificación [23].

La división se realiza por medio de la frontera conocida como hiperplano. Sea una línea en un espacio bidimensional, o un plano en uno tridimensional, éste separa los datos en función de las categorías elegidas, las etiquetas necesarias en este método. No sólo se crea una división, sino también un margen alrededor de ella que es máximo en el sentido de los datos de cada clase. Para el cálculo de la frontera, únicamente se tienen en cuenta las posiciones de los datos más próximos a la división, aquellos que resultan conflictivos, a diferencia de la regresión lineal, que minimiza el error con todos los datos disponibles.

En otros casos en los que una recta no es suficiente para separar dos conjuntos por sus características, se recurre a técnicas no lineales para establecer estas fronteras, funciones de los datos que se quieren comparar, para las que es necesaria la función "Kernel". [24]

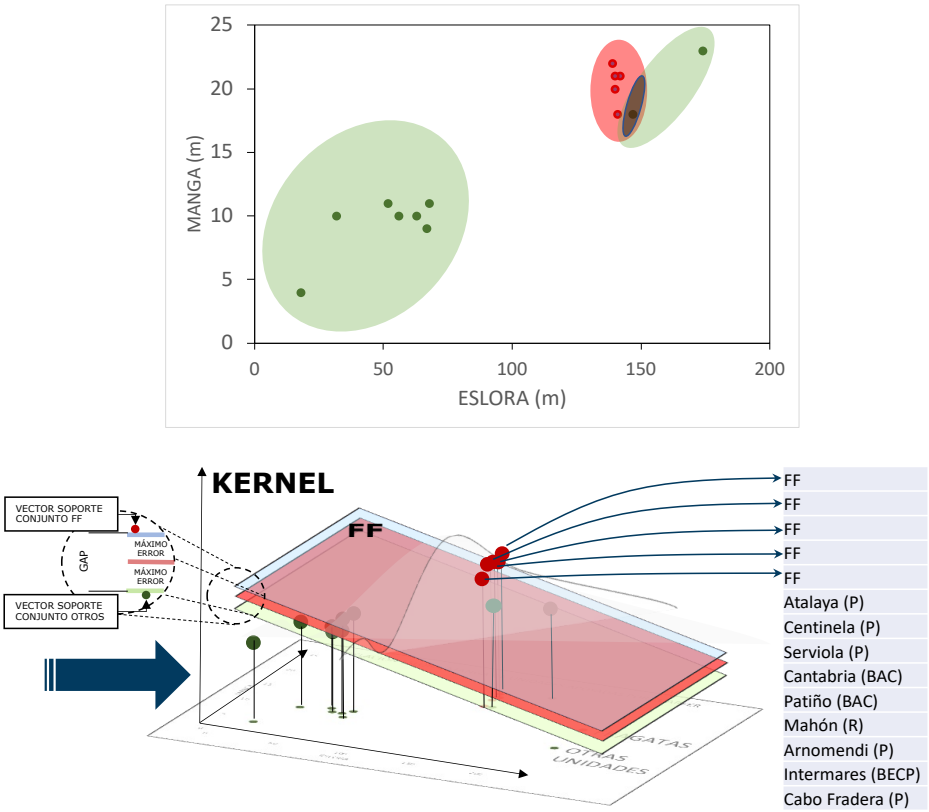


Figura 2-9. Ejemplo del empleo de SVM para clasificación de buques tipo FF
Elaboración propia

En el caso presentado por la Figura 2-9, los buques con apoyados por el Arsenal de Ferrol son clasificados en función de su eslora (eje abscisas) y manga, (eje ordenadas) como se presenta en el gráfico superior. Para diferenciar aquellos que son fragatas (FF) de otras unidades, se aplica la herramienta del Kernel. Se emplea la tercera dimensión en el eje z, posicionando los puntos sobre una función que es parámetro de los otros dos, siendo de máximo valor en la eslora de 140 metros. Los vectores soporte, o puntos más cercanos a la frontera, serán la representación del Patiño como punto del conjunto negativo (otros) y cualquiera de las F-100 como puntos del conjunto de fragatas.

2.9.2.4 Árboles de decisión

Los árboles de decisión (decision trees) son una técnica de aprendizaje supervisado que permite llevar a cabo pruebas dentro de los datos de un conjunto para alcanzar una salida final. Como su nombre indica, se basa en un algoritmo de toma de decisiones en cada una de las ramas que emana del árbol principal. Estas ramas y sus terminaciones (hojas) deben tener en cuenta todos los factores que conciernan a la toma del camino adecuado en función de la situación a desarrollar. Dependiendo de la naturaleza de la decisión, se distingue entre varios tipos de árboles: [25]

- Árboles de clasificación.
- Árboles de regresión.
- Bosques de árboles de decisión.
- Árboles de clasificación y regresión.
- K-means clustering.

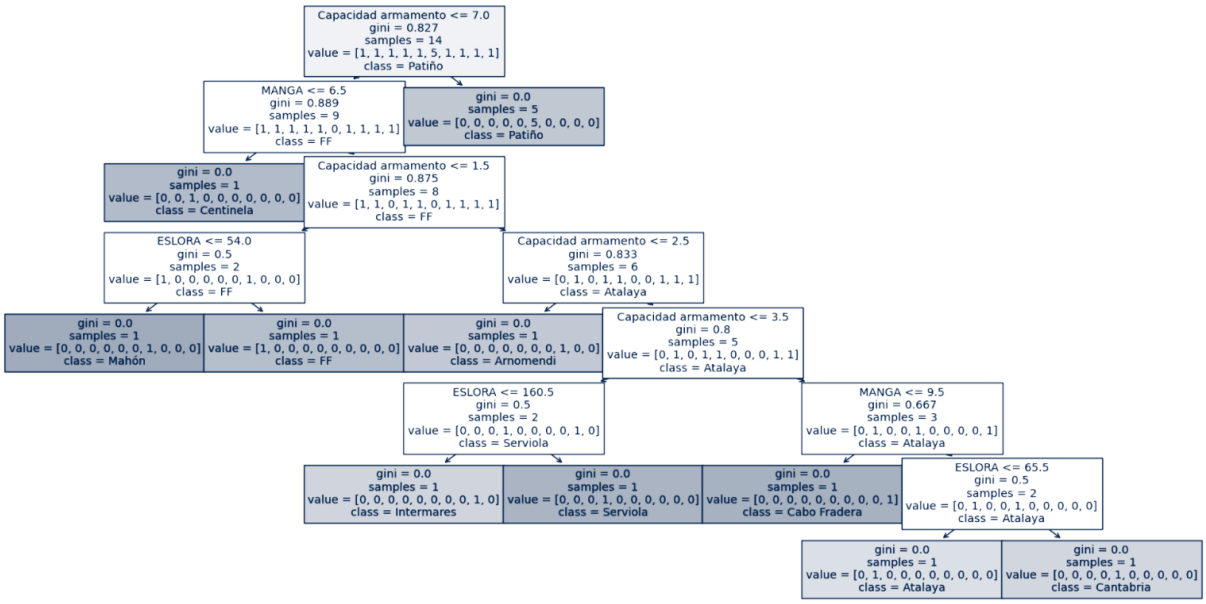


Figura 2-10. Árbol de decisión buques ARFER
Elaboración propia

La Figura 2-10 muestra un ejemplo de árbol de decisión de clasificación elaborado con la librería de Python de *scikit-learn*. En él, se muestra el mismo conjunto de datos que en la Figura 2-10 añadiendo la variable de la capacidad armamentística del buque.

A la hora de llevar a cabo la división de las variables por sus clases, se pueden seleccionar métodos diferentes para hacerlo, como el método del Gini y el criterio de la entropía.

- El método del Gini mide, en términos de probabilidad, la pureza de un nodo principal. Esta probabilidad es la de no tener un mismo nodo dos clases iguales en su interior, cuanto menor sea el valor del Gini, mayor será la pureza. Como medida de probabilidad, su valor está

comprendido en el conjunto [0,1]=$\{x \in \mathbb{R}: 0 \leq x \leq 1\}$. El valor del Gini se obtiene de la ecuación (10).

$$GINI(t) = 1 - \sum_{i=1}^{n} (P_i)^2 \qquad (10)$$

Siendo P_i la probabilidad de que una muestra sea de la clase i.

- La entropía es una expresión empleada para medir el desorden dentro de un sistema. En el caso de tener un nodo puro, la entropía será nula. Si por el contrario se repiten las clases con la misma frecuencia, la entropía será máxima.

$$H = -\sum_{i=1}^{n} P_i \cdot \log_2 P_i \qquad (11)$$

2.9.2.5 Random forests

Random forests es un algoritmo de aprendizaje supervisado que se basa en la construcción de un modelo a partir de distintos árboles de decisión, explicados en el subapartado 2.9.2.4. Esta unión de varios modelos en uno solo se conoce como técnica de "ensemble". Su diferencia con el empleo de múltiples árboles de decisión radica en la aleatoriedad para el establecimiento de nodos raíz y la segregación del resto de nodos. Esto se conoce como método de Bagging, o Bootstrap aggregation [26].

Más en detalle, existen dos técnicas de "*ensemble*", "*bagging"* y "*boosting"*. El bagging utiliza una metodología de combinación de algoritmos en paralelo, teniendo como salida final la opción más repetida. El caso del boosting, sigue un proceso secuencial en el que los modelos más "débiles" recurren a los resultados de los más fiables para generar un modelo más preciso.

De vuelta a random forests, el proceso de bagging selecciona paquetes de datos aleatorios para generar los árboles de decisión de cada sub-modelo, de forma que se garantizan entrenamientos independientes para cada rama, conocido como Bootstrap. Finalmente, se obtienen todos los *outputs* y, la opción más repetida, será el *output* global, lo que se conoce como *aggregation*.

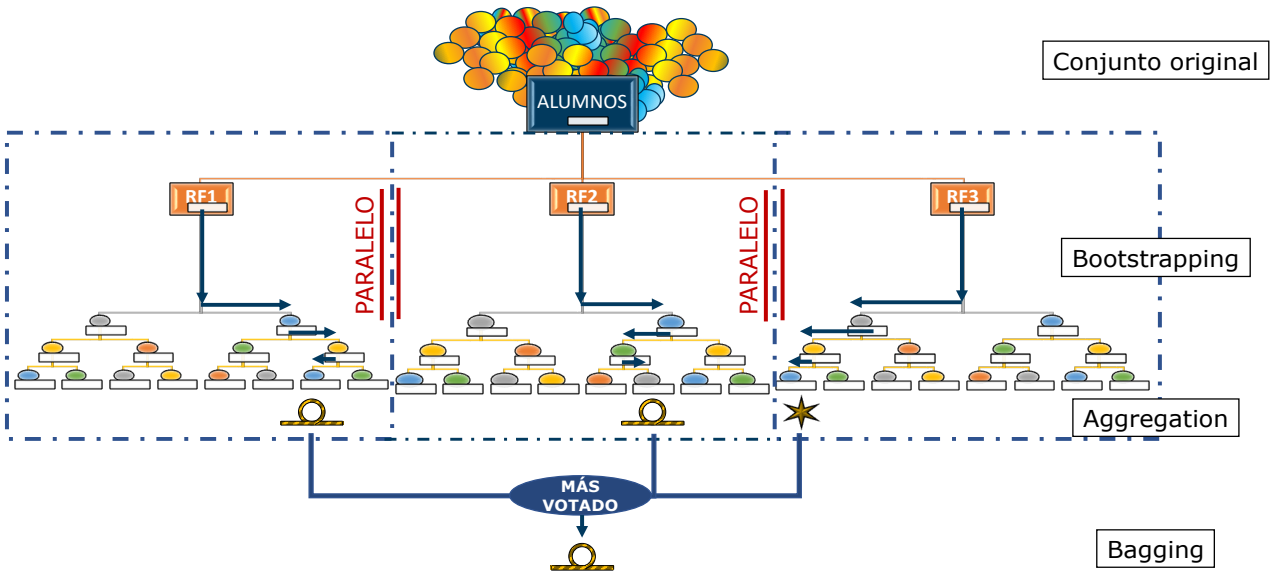


Figura 2-11. Técnica Random Forests
Elaboración propia

En el ejemplo de la Figura 2-11, se puede ver como una muestra original de todo el alumnado de la Escuela Naval es dividido en 3 árboles de decisión distintos, que trabajan en paralelo hasta llegar a una decisión final, el proceso de *bootstrapping*. Evaluando los resultados individuales con sus equivalentes, *aggregation*, dos tercios de las votaciones seleccionan la opción de cuerpo general, por lo que a la hora del *bagging*, es el elegido por el modelo conjunto.

La regresión es otra de las técnicas que random forests es capaz de llevar a cabo. En esta modalidad, cada uno de los árboles realiza una predicción específica, y la media de todos estos outputs será el valor final de la predicción del modelo completo. Esto es lo que lo diferencia del mecanismo de clasificación de random forest, cuya salida es determinada por la clase de los árboles de decisión.

2.9.2.6 Redes neuronales

Una red neuronal es un conjunto de algoritmos cuya función es la de detectar, localizar y explotar las relaciones existentes en un conjunto de datos de una forma similar a la realizada por un cerebro humano. Su modo de trabajo incluye la posibilidad de variar los inputs del conjunto para obtener un resultado lo más aproximado al valor esperado [23]. Son también conocidas como redes neuronales artificiales (ANN) o redes neuronales simuladas (SNN) y se estructuran al más alto nivel en tres grupos de elementos:

- Capa de entrada: en la que se reciben los datos que serán procesados por la red.
- Una o varias capas ocultas: son aquellas encargadas del aprendizaje de la red, con el mayor valor computacional del sistema.
- Capa de salida: está formada por las neuronas que determinarán el valor de salida de la red.

Cada una de estas divisiones contiene en su interior un número de nodos, conectados entre sí y que tienen asociado un peso y un umbral de activación que, en función del valor a la salida del nodo, determinará si se seguirá el camino a través de este hacia los nodos contiguos. En cada nodo se cuenta con un modelo de regresión lineal independiente de los demás, en el que actúan un valor de entrada, ponderaciones, un sesgo que corrija la variabilidad del cálculo, y un valor de salida [27].

$$\sum_{i=1}^{n} w_i x_i + sesgo \tag{12}$$

$$salida = f(x)\begin{cases} 1\ si\ \sum_{i=1}^{n} w_i x_i + sesgo\ \geq 0 \\ 0\ si\ \sum_{i=1}^{n} w_i x_i + sesgo\ < 0 \end{cases} \tag{13}$$

En la ecuación (13) se muestra el output de un nodo, un valor discreto del conjunto [0,1]=$\{x \in \mathbb{Z}: 0 \leq x \leq 1\}$. Para decidir el valor de la salida, se hace uso de lo que se conoce como funciones de activación, que transforman el comportamiento lineal de la ecuación (12), lo que conseguirá un mayor ajuste de los resultados a los que generaría una neurona real. Los ejemplos más empleados son la función escalón y la función logística.

- Función escalón: esta función devuelve el valor "0" para todos los inputs negativos y el valor "1", para los positivos. No es apta para el aprendizaje automático actual ya que las correcciones en redes neuronales se realizan a partir de la derivada de la función de costo capa por capa, conocido como "*backpropagation*". La función escalón no es diferenciable ya que su derivada en el conjunto $x \in \mathbb{R}: \{-\infty < x < 0\} \cup \{0 < x < +\infty\}$ será igual a 1 siempre, mientras que en $x = 0$, no existe.

- Función logística: sus outputs están acotados entre 0 y 1, y es altamente empleada en algoritmos de clasificación binaria. Está definida por la ecuación (14), y se distribuye conforme se puede ver en el gráfico de la Figura 2-12. Los beneficios que aporta, a diferencia de la función escalón, son una corrección adecuada de los pesos mediante la técnica de "backpropagation", aplicando correcciones acusadas en las proximidades a la abscisa "0". [28]

$$y = \frac{1}{1 + e^{-x}} \qquad (14)$$

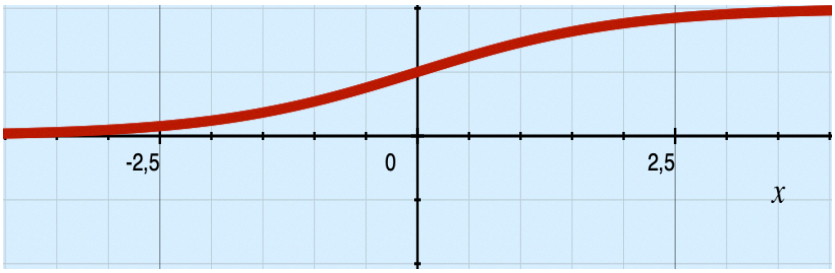


Figura 2-12. Función logística
Elaboración propia

2.9.2.7 Deep Learning

Después de presentar los principios de las redes neuronales, es posible caracterizar el aprendizaje profundo esencialmente como una ANN de tres o más capas. Lo que le distingue de los ámbitos más convencionales del ML es el tipo de datos con los que trabajan sus algoritmos. Los conjuntos de datos a analizar no reciben ningún tipo de pretratamiento, como sí acostumbraban en las técnicas convencionales.

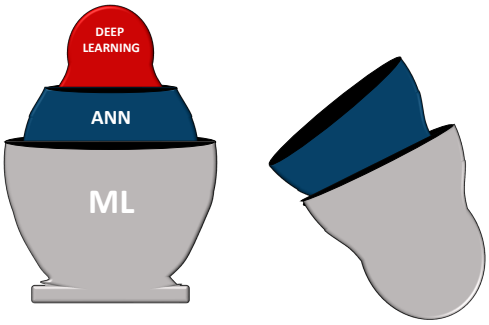


Figura 2-13. Tecnologías dentro de ML
Elaboración propia

El Deep learning facilita los trabajos previos al estudio de datos con la automatización de procesos necesaria, lo que lo convierte en una de las herramientas más válidas para trabajar con datos que no siguen estructuras lógicas o que carecen de valores en ciertas áreas [29].

2.10 Lanchas Rodman 66

Para llevar a cabo el desarrollo de este proyecto, un posible método de validación es la prueba en unidades operativas actualmente. La Escuela Naval Militar de Marín dispone actualmente de cuatro lanchas de instrucción (LLII) de la clase Rodman 66 construidas en los astilleros del grupo homónimo.

Son embarcaciones fabricadas en poliéster reforzado en fibra de vidrio (PRFV), con casco en forma de v hacia la proa, que en la *publicación 4810 de la Escuela Naval Militar* se definen con las siguientes características generales [30]:

Característica	Valor	Característica	Valor	Característica	Valor
Eslora total	20,5 m	Eslora del casco	20 m	Velocidad máxima	23 kts
Manga Total	5,1 m	Manga del casco	4,9 m	Velocidad de crucero superior a	15 kts
Puntal	2,56 m	Calado aprox. en rosca	0,9 m	Desplazamiento en rosca	29,5 Tn
Capacidad de combustible	5000 [4]L	Capacidad de agua	500 L	Tripulación prevista	16 pax
Potencia máxima en servicio de prácticos	2 x 715 CV	Propulsión	Ejes y hélices	Alojamiento	16 pax

Tabla 2-3. Datos generales de la Pub. ENM 4810 [30]

2.10.1 Sistema propulsor

El sistema propulsor de cada lancha se compone de dos motores diesel CATERPILLAR C18 y dos reductores hidráulicos ZF550. Los motores tienen refrigeración por medio de un circuito abierto de agua salada. La información de operación de ambos motores se presenta en sus respectivos paneles de monitorización Caterpillar Marine Power Display (MPD), con pantalla LCD y posibilidad de conexión a software especializado de CAT para la obtención de datos mediante el adaptador de comunicación CAT COMM III. Los datos son presentados de forma simplificada por medio de simbología como se muestra en la Tabla 2-4 con los datos de mayor interés a bordo de las lanchas de instrucción. La lista completa de variables que proporciona el MPD se encuentra disponible en su guía de instalación en la tabla 10-2 [31].

Battery Voltage		Fuel Rate	
Engine Load		Oil Pressure	
Transmission Temperature		Boost Pressure	
Engine Speed		Coolant Temperature	
Transmission Pressure			

Tabla 2-4. Leyenda de simbología CAT Marine Power Display
Elaboración propia en base a **[31]**

[4] La A-123 "GM Rull" y A-124 "GM Salas" tienen capacidad para 6000 y 7000 litros respectivamente.

2.11 Trabajos previos en el ámbito de estudio

A continuación, se listan algunos trabajos que, por su relación con el presente, han sido considerados de interés.

2.11.1 Training Boat Digital Twin

El proyecto TRABODIT (Training Boat Digital Twin) es uno de los trabajos de investigación desarrollados por miembros del Centro Universitario de la Defensa de la Escuela Naval Militar[5]. En él, se pretende establecer los procedimientos necesarios para el desarrollo de gemelos digitales a partir de las unidades basadas en la ENM y que son también objeto de este trabajo, las lanchas de instrucción Rodman 66.

El enfoque de este proyecto comprende aspectos de mantenimiento, formación y seguridad entre otros, centrándose en la simulación y desarrollo de modelos de operación de esta plataforma. Para ello, se plantea la sensorización de las lanchas y el tratamiento del dato mediante técnicas de machine learning. En una segunda etapa de desarrollo, se pretende integrar los modelos obtenidos en el entorno del simulador de navegación (SIMNAV) de la ENM, como herramienta para el adiestramiento de los alumnos. [32]

2.11.2 El gemelo digital como base tecnológica para el desarrollo logístico de la Armada 4.0

Se trata de un trabajo de fin de grado realizado por el entonces alférez de fragata Alejandro Pérez Seoane en el que se estudian las mejoras que implementaría al sistema logístico de la Armada la inclusión de los gemelos digitales.

Es un trabajo que realiza una extensa revisión bibliográfica en los aspectos relacionados con la virtualización del entorno físico. Como resultado, se establecen las implicaciones que tiene la transformación digital en el ámbito de la industria militar [33].

2.11.3 Ghenova

A pesar de no ser un proyecto, sino una empresa de ingeniería, su labor en el ámbito de la transformación digital en la industria de defensa la convierte en objeto de estudio para este trabajo. Entre sus áreas de negocio, dentro del naval militar se encuentra el programa F110, cuyo cliente es Navantia y cliente final la Armada.

El tipo de asistencia que presta Ghenova incluye el diseño básico de equipos, sistemas mecánicos, electrónicos y combate, ingeniería de apoyo logístico integrado (ILS) hasta la revisión crítica de diseño, soporte en la integración de sistemas y, la más reseñable, la incorporación del gemelo digital.

El gemelo digital de la F110 se encuentra en el marco del programa de nivel superior READY-TWIN interrelacionando las herramientas de inteligencia artificial, IoT, realidad virtual y realidad aumentada. La aplicabilidad de este desarrollo es de mayor alcance en el ámbito del mantenimiento de acuerdo con las necesidades de la DISOS; como se explica en el subapartado “El gemelo digital en la Armada”. [34]

[5] Conforme al RD 1723/2008 de 24 de octubre, los Centros Universitarios de la Defensa tienen entre sus fines, recogidos en el artículo 4, la definición y el desarrollo de líneas de investigación que sean consideradas de interés para las Fuerzas Armadas, la paz, la seguridad y la defensa.

3 DESARROLLO DEL TFG

3.1 Generalidades

Este capítulo contiene toda la información relativa al avance del trabajo realizado. En él, se trata de mantener una secuencia cronológica de las tareas completadas, a pesar de existir solapes temporales entre sí, como se detalla en el apartado de planificación.

También, se darán en algunos ámbitos justificaciones para la toma de decisiones en líneas de acción a seguir en ciertos trabajos, todas ellas justificadas conforme a la normativa PECAL 2070 y UNE-ISO 31000 explicado en el subapartado 3.4.

3.2 Reparto temporal de Tareas

3.2.1 Planificación

La primera tarea realizada es la planificación temporal del trabajo, así como la asignación de recursos para cada uno de los trabajos. Este reparto está disponible para su consulta y mayor detalle en el Anexo IV: Planificación temporal del trabajo, desarrollado en el software MS Project.

Para el reparto, se ha dividido el proyecto por áreas, de investigación y consulta, documentación, formación y materialización.

La primera etapa comprende la revisión bibliográfica para llevar a cabo tanto la ambientación al trabajo como la selección de fuentes de información en las que se puedan apoyar los procesos de toma de decisiones. Se ha diferenciado entre la investigación propia de fuentes disponibles en medio físico o digital de las entrevistas con expertos. De estas últimas, se ha concertado un encuentro con el jefe del CESADAR periférico del Arsenal de Ferrol para obtener información de primera mano del funcionamiento de la aplicación ATAVIA. También una con el Jefe del Centro de Inteligencia Artificial de la Armada, para conocimiento de las líneas futuras de integración de las lanchas de instrucción, tanto las actuales como las 2 unidades que se recibirán en el año 2023, en el programa de mantenimiento predictivo del CESADAR con técnicas de IA. Dentro de la revisión, se contempla un proceso periódico de información, para cubrir las necesidades que surjan durante la ejecución de las posteriores tareas.

Existe otra fase de búsqueda de información pero que ha sido separada por sus menor grado de especificidad comparada con la revisión bibliográfica. Se trata de los procesos formativos para obtener las competencias básicas para el empleo de los diferentes software, MS Excel, VS Code programando en lenguaje Python y Unreal Engine.

A continuación, se planifican la documentación de procesos en esta memoria, así como otras entregas a realizar de carácter expositivo. Este periodo documental se realiza en paralelo con el desarrollo del proyecto, y mantiene dentro de él relaciones Fin-Comienzo y Fin-Comienzo+1.

Tras su análisis, las tareas críticas que se han detectado son:

- Redacción de los resultados/ Validación/Pruebas.
- Redacción del apartado conclusiones y líneas futuras.
- Redacción de los anexos del sello de calidad EUR-ACE (sobre la concienciación social, económica y medioambiental de este trabajo)
- Redacción del resumen.

Para su correcto tratamiento, se hará una redacción adelantada de estos apartados, compaginándolos con otras tareas para asegurar que, debido a su proximidad con la fecha de entrega del trabajo, no se produzcan retrasos al desarrollo general.

Los hitos que se han definido durante el proyecto son los correspondientes a las entregas oficiales de acuerdo con las evaluaciones de los tutores del trabajo y otros puntos de referencia establecidos en la guía docente del proyecto.

3.2.2 Fuentes de información

La principal fuente de información para la realización del trabajo, han sido repositorios on-line de documentación académica. No obstante, también se han realizado entrevistas con personal experto, como aparece planificado en el reparto de tareas, en el periodo del primer mes de trabajo. El personal entrevistado forma parte de organismos que trabajan en los actuales proyectos en desarrollo relacionados con la digitalización de la Armada, como el jefe del CESADAR periférico de Ferrol y el Jefe del Centro de Inteligencia Artificial de la Armada.

Los datos empleados para el entrenamiento de los modelos y validación del programa han sido obtenidos de la plataforma ATAVIA de la fragata F-105 “Cristóbal Colón”, de la clase Álvaro de Bazán.

Los datos de las lanchas de instrucción han sido obtenidos a bordo de estas durante los periodos de adiestramiento a flote acordados.

La validación de las fuentes de información se ha realizado conforme a la metodología OPVL[6] desarrollada por el International Baccaleurate Organization dividiendo el origen en tres tipos: fuentes militares, fuentes de universidades y académicas y sitios web [35].

- Las fuentes militares son aquellas que provienen del Área del Repositorio Corporativo de la Armada (ARCA) o de la red intranet de la Armada y que se encuentran dentro de los tipos doctrina, píldoras docentes de la Unidad de Enseñanza Virtual Corporativa de la Armada (UVICOA), instrucciones permanentes (IP), conceptos de mandos (CM) y documentación industrial (DI).
- Las fuentes de universidades y académicas han sido obtenidas de plataformas como del repositorio del Institute of Electrial and Electronics Engineers (IEEEXplore) y otros de universidades, habiendo documentos de los tipos actas de conferencias (AC), artículos (AR) y trabajos universitarios (TU) entre la revisión bibliográfica del trabajo.
- Las fuentes de sitios web (SW) han sido consultadas para conocimiento de los software en sus páginas oficiales, evitando fuentes de terceros.

[6] Técnica del IBO que evalúa la fiabilidad de las fuentes en función de su origen (O), propósito de empleo (P), valor de contenido (V) y limitaciones (L) en cuanto a la información obtenida.

FUENTE		ORIGEN	PROPÓSITO	VALOR	LIMITACIONES
FUENTES MILITARES	ARCA	Intranet Doctrina firmada por miembros JUTAC en el Área del Repositorio Corporativo de la Armada.	Conocimiento de los sistemas actuales y programas en el ámbito de la Armada.	Doctrina única y común a toda la Armada. Manuales desarrollados por personal experto en la materia.	Accesibilidad limitada a acceso a Intranet. Limitación por Habilitación Personal de Seguridad (HPS) en función de la Clasificación de seguridad. Información objetiva de carácter doctrinal en materia de defensa.
	UVICOA	Intranet. Ratificado por DIENA	Conocimiento del SICP, funcionamiento e interfaz del operador.	Contenidos de formación de especialistas en el ámbito correspondiente. Conocimientos detallados.	Accesibilidad limitada a acceso a Intranet. Información objetiva en materia de formación en el ámbito de la Armada.
	IP	Ratificado por mando correspondiente. Empleadas las instrucciones del AJAL	Conocimiento de las órdenes en el ámbito del Apoyo Logístico en la Armada. Requisitos de las herramientas logísticas	Órdenes comunes a todas las UCOs afectadas. Criterio único para toda la Armada.	Accesibilidad limitada a acceso a Intranet. Órdenes precisas, no ambigüedades.
	CM	Ratificado por mando correspondiente. Empleado el Concepto de apoyo logístico y Concepto del arsenal 4.0	Generalidades de las ideas sobre futuros cambios y transformación digital en la Armada.		Accesibilidad limitada a acceso a Intranet. Órdenes precisas, no ambigüedades.
	DI	Indra Sistemas. Disponible en Intranet	Presentación de aplicaciones de mantenimiento predictivo y análisis de datos	Información de primera mano sobre las herramientas desarrolladas.	Accesibilidad limitada a acceso a Intranet. Sujeta al interés industrial de la empresa.
FUENTES UNIVERSITARIAS	AC	IEEEXplore	Adquisición de conocimientos en la temática del trabajo.	Actas oficiales de conferencias, de organismos reconocidos. Alta fiabilidad.	Información de expertos, de mayor objetividad en las
	AR	Elsevier			Información subjetiva, basada en documentación fiable. Se muestra el punto de vista del autor en algunos casos.
	TU	Universidad correspondiente		Información técnica especializada. Precisión y fiabilidad.	Información de trabajos/proyectos de fin de grado.
SITIOS WEB	SW	Página oficial del software. Unreal Engine; Librerías de Pyhton; VS Code; Jupyter.	Conocimiento generalidades y funcionamiento del software a emplear.		No son documentación oficial. Volátil por su ubicación en línea.

Tabla 3-1. Análisis OPVL
Elaboración propia

3.3 Descripción del entorno de trabajo

3.3.1 Miscrosoft Excel

Los datos primarios de ATAVIA son descargados en un archivo de formato CSV (Comma Separated Values). Con el fin de obtener una primera impresión de su naturaleza para su posterior manipulación y

análisis de correlaciones, han sido tratados por el software de hojas de cálculo MS Excel, también empleado como registro para la base de datos.

MS Excel es un software de hojas de cálculo desarrollado por Microsoft en 1985 como evolución de su anterior software "Multipan". Permite el análisis de datos con diferentes funciones matemáticas y herramientas variadas. Es apto para gran variedad de ámbitos gracias a la capacidad multiplicadora que ofrece la programación integrada de "Visual Basic", que incluye la automatización de procesos por medio de macros.

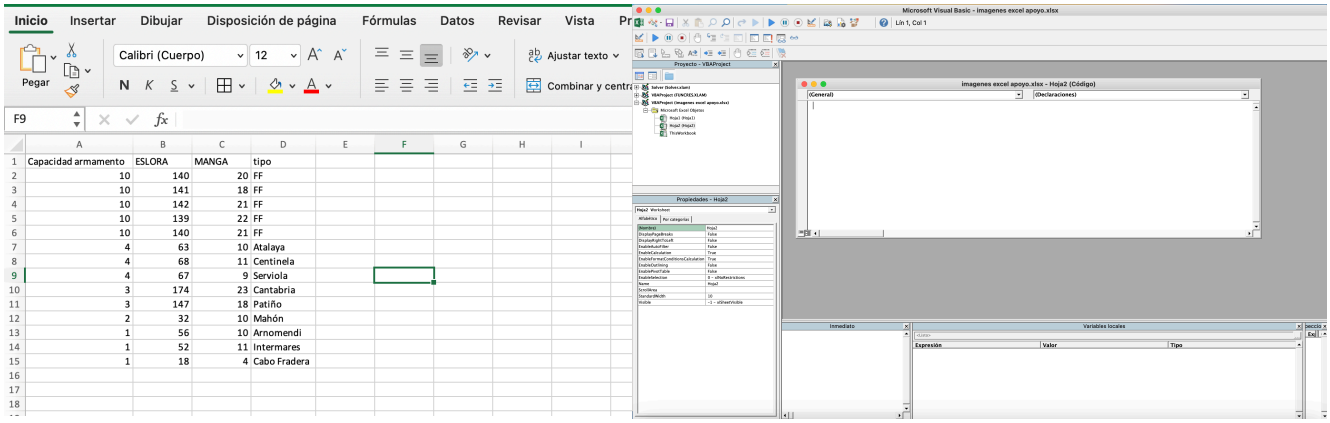


Figura 3-1. Interfaz MS Excel y Visual Basic
Elaboración propia

3.3.2 Python

Python es un lenguaje de programación de alto nivel inventado en 1991 y que ha sido actualizado a lo largo de los años hasta llegar a su versión 2.7, lanzada en 2015, la cual ofrece grandes prestaciones. Se trata de una herramienta multipropósito que, con las librerías adecuadas, permite ejecutar scripts para una gran abanico de aplicaciones. Sus principales ventajas en el desarrollo son el ser un lenguaje interpretado y dinámico, ejecutándose el código fuente instrucción a instrucción, y actualizando las variables según lo marcado por el script [36].

En este trabajo, se emplea la versión Python 3.11.2, lanzada en febrero de 2023, que ofrece mejoras con respecto a la versión anterior en velocidad de procesado [37].

3.3.2.1 Visual Studio Code

Para llevar a cabo la tarea de programar, una de las herramientas más útiles es un entorno de desarrollo (IDE por sus siglas en inglés), empleado para escribir y compilar código [38]. Visual Studio Code (VS Code) es una IDE disponible como aplicación de escritorio para sistemas operativos Windows, MacOs y Linux.

En este trabajo, se emplea VS Code como plataforma para el desarrollo de la aplicación de Streamlit, en la que se integran todos los componentes del modelado de los buques. Se ha empleado para el desarrollo la extensión de Jupyter Notebooks, explicada con mayor detalle en el apartado 3.3.2.2.

La decisión de empleo de VS Code como IDE se basa en la posibilidad que ofrece de trabajar con lo que se conoce como un "workspace system", o un directorio independiente para manejo de archivos contenedores del código fuente (y dependientes), y otro, para la lectura y ejecución de dichos códigos. De esta forma, VS Code relega el conocimiento de los lenguajes de programación en las extensiones que elija personal programador dentro de su interfaz. [39]

3.3.2.2 Jupyter Lab

De acuerdo con lo expuesto en el subapartado de Planificación, la primera fase del trabajo comprende una etapa de formación para el manejo de los programas y lenguajes de programación. El entorno elegido para el aprendizaje de Python, así como el pretratamiento de datos es la extensión Jupyter Lab dentro de la IDE VS Code.

Jupyter Lab es una aplicación web que cuenta con una interfaz flexible que permite al usuario la personalización del flujo de trabajo y variadas opciones de interacción. Es un desarrollo de la organización Proyecto Jupyter, evolución de otro proyecto llamado Jupyter Notebook, de prestaciones similares, y que con el tiempo será reemplazado por éste. Otra aplicación disponible en el entorno Jupyter es Jupyter-Hub, un entorno colaborativo en un servidor proxy multiusuario, lo que facilita los trabajos conjuntos en la plataforma.

El programa es ejecutado desde navegadores sobre una base de código abierto con diferentes opciones de exportación, lo que hace ver su versatilidad como herramienta. Para su manejo, el usuario introduce el bloque de código a ejecutar en su correspondiente espacio, pudiendo ejecutarlo a continuación y volver a escribir en un nuevo espacio de trabajo que aparece en pantalla. [40]

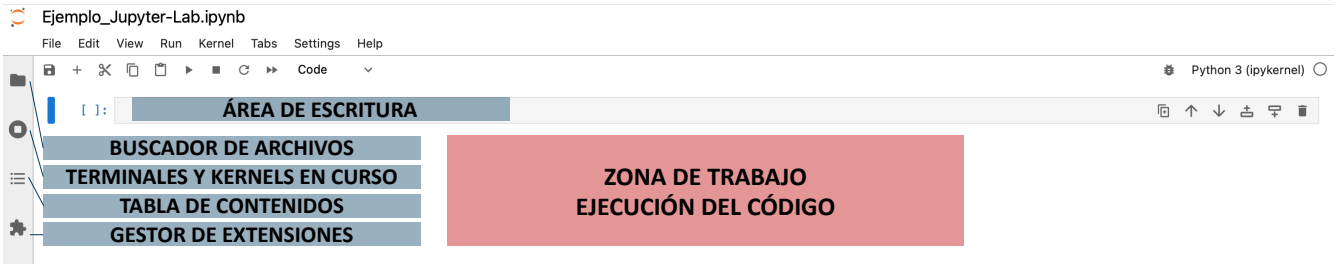


Figura 3-2. Interfaz de Jupyter Lab
Elaboración propia

La interfaz con la que se ha trabajado en este desarrollo es la de la extensión en VS Code, como se ha mencionado anteriormente. La Figura 3-3 muestra dicha interfaz, que se presenta como una réplica de lo que se puede apreciar en la aplicación original de Jupyter.

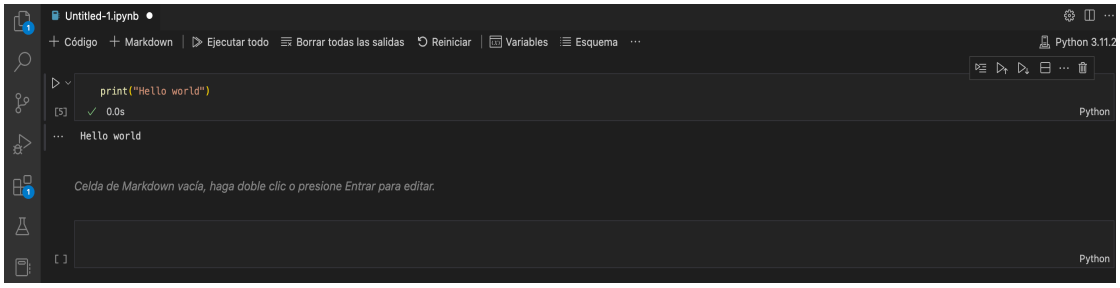


Figura 3-3. Interfaz de la extensión Jupyter Notebook en Visual Studio Code
Elaboración propia

3.3.2.3 Librerías de Python

Las librerías elegidas para emplear en este trabajo deben ser capaces de procesar datos, ser aptas para análisis y representación de los modelos obtenidos. Es por ello por lo que se han definido las siguientes necesidades a cubrir:

- Tratamiento de hojas de cálculo y formatos similares.
- Desarrollo de una GUI (Graphical User Interface).
- Análisis de los datos.

- Representación gráfica de los datos.

Cada una de estas necesidades, por su entidad, ha de ser cubierta en su totalidad por las librerías seleccionadas, por lo que se ha realizado la siguiente selección:

- Pandas: la librería Python for Data Analysis (PANDAS) permite el tratamiento de datos que estén contenidos en lo que se conoce como "dataframes", como archivos en formatos CSV, MS Excel, SQL y HDF5. Su misión se resume en el tratamiento de estos datos como elementos de una tabla una vez se importan al archivo de Python. [41]
- Streamlit: se trata de una librería creada en 2019 como un facilitador para los desarrolladores de aplicaciones relacionadas con machine learning para disponer de una herramienta de desarrollo de aplicaciones interactuables con sus modelos. [42]
- Sci-kit learn: se trata de la librería más popular de Python para Machine Learning gracias a su variedad de algoritmos integrados en ella.
- Matplotlib: es una librería de representación de gráficos similares a los que genera el programa MATLAB, ofreciendo variadas opciones de personalización. [43]

3.3.3 Unreal Engine 5

Unreal Engine es el motor gráfico desarrollado por la compañía Epic Games. Es una herramienta adecuada para proyectos de simulación que precisen de realismo, así como los que integran la realidad virtual.

La programación dentro de Unreal es dual ya que es posible realizarla tanto en lenguaje C++ como por medio de un sistema multinodal basado en blueprints. Este último es el sistema elegido para el trabajo, ya que reduce los tiempos necesarios de formación (interfaz visual frente a una alfanumérica).

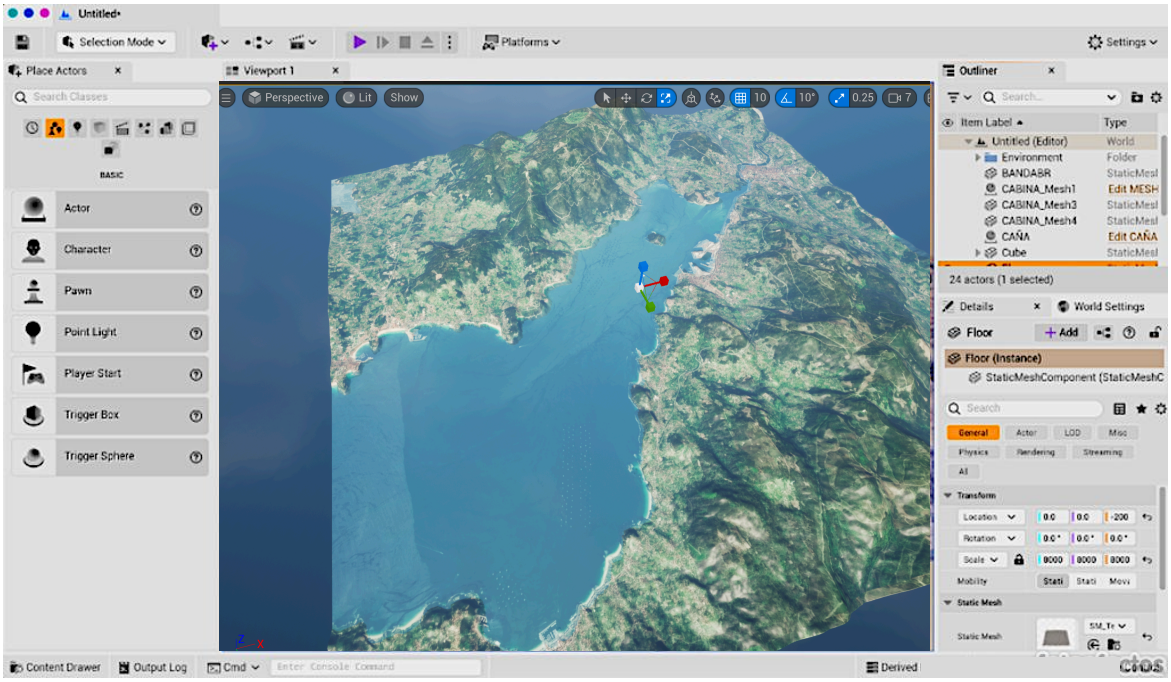


Figura 3-4. Entorno Unreal Engine
Toma en desarrollo de la ría de Pontevedra. Elaboración propia

3.4 Gestión de riesgos

A lo largo del trabajo, se ha llevado a cabo una misma metodología para la toma de decisiones basada en los riesgos inherentes a las posibles líneas de acción. Ésta forma de trabajo se basa en las normativas UNE-ISO 31000 y PECAL 2070, de directrices de gestión de riesgos y calidad OTAN respectivamente.

De acuerdo con la normativa ISO, se definirán las fuentes de riesgo, como potenciales generadores del mismo, los eventos, como ocurrencias o cambios de un conjunto particular de circunstancias, y las consecuencias, como el impacto en el proyecto.

Los riesgos serán evaluados conforme a su probabilidad y su impacto de acuerdo con la Tabla 3-2. En materia de riesgos, se entiende probabilidad no como un valor numérico estadístico sino como la posibilidad de su ocurrencia del evento. La normativa PECAL habla de tres niveles de probabilidad; alto, medio y bajo, y detalla sus características en el anexo C de dicha publicación. La normativa ISO, en cuanto al empleo de matrices de consecuencia/probabilidad, habla de la validez de cualquier tipo de escala [44] [45].

Probabilidad \ Impacto	1	2	3	4	5
5	5	20	45	80	125
4	4	16	36	64	100
3	3	12	27	48	75
2	2	8	18	32	50
1	1	4	9	16	25

Tabla 3-2. Matriz de gestión de riesgos empleada
Elaboración propia

Se ha decidido valorar con un mayor peso en la matriz el impacto en el trabajo que su probabilidad de ocurrencia, siguiendo la fórmula:

$$C_{ij} = p \cdot i^2 \qquad (15)$$

Siendo "p" la propabilidad de ocurrencia e "i" el impacto que conllevaría.

La razón para el empleo de esta metodología se basa en su empleo en el ámbito del Ministerio de Defensa, en especial por los Representantes para el Aseguramiento de la Calidad del MINISDEF (RAC) y, sobre todo, en la gestión de programas de la DGAM.

3.5 Minería de datos y análisis

En este apartado se justifican los datos que se van a tratar con el programa de monitorización, así como el desarrollo de los algoritmos empleados para el tratamiento de los datos.

3.5.1 Selección de variables

A fecha de marzo de 2023, los buques de la Fuerza de Combate de Superficie (FUCOM) son unidades propulsadas eminentemente por motores diésel. Es por ello, que se han decidido tomar datos relativos al funcionamiento y operación de los propulsores.

Como plataforma de validación se han elegido las lanchas de instrucción, gracias a su proximidad y accesibilidad para la toma de datos y replicado. Al contrario que los buques monitorizados por ATAVIA, las lanchas de instrucción basadas en la ENM no cuentan actualmente con una base de datos que se actualice con cierta continuidad. Sí que se realiza durante sus periodos de mantenimiento recopilación de historiales de los motores con el CAT COMM III por parte de Rodman, así como de otros datos cinemáticos con un conector RJ45 desde una toma debajo de las consolas de la banda de estribor en el puente. Al no disponer la Escuela Naval del material necesario para realizar una transferencia

automática, los datos son tomados en vídeo desde los MPD del puente (ver apartado 2.10.1). Las variables medidas son las mismas que se extraen con el CAT COMM III, por lo que no hay pérdida de información en ese sentido, pero sí será necesario hacer un tratamiento para obtener una mayor cantidad de datos para entrenamiento de los modelos (por ejemplo, con técnicas de oversampling o interpolaciones lineales entre dos variables separadas un tiempo conocido).

Las variables que medir elegidas son las de la Tabla 3-3:

Nomenclatura CAT	Nomenclatura ATAVIA (F-105)	TAG NAME ATAVIA (F-105)
Engine Load	Posición de la cremallera (%)	DE1.fuelrack DE2.fuelrack
Transmission Temperature	Indicador y alarma de alta temperatura sección de popa del cojinete del eje del M.P.	GB1.S270.temperature GB2.S270.temperature
Engine Speed	Velocidad del motor (RPM)	DE1.speed DE2.speed
Fuel Rate	Algoritmo combustible consumo medio	DE1.flow DE2.flow
Oil Pressure	Presión aceite (BAR)	DE1-LUB.oil.pressure DE2-LUB.oil.pressure
Coolant Temperature	Temperatura agua refrigeración salida	DE1-WAT.ac.oc.outlet.temperature DE2-WAT.ac.oc.outlet.temperature
Transmission Pressure	Indicador y alarma de baja presión en el freno de los ejes	GB1.S210.pressure GB2.S210.pressure

Tabla 3-3. Variables medidas en este trabajo
Elaboración propia

Sobre la tabla anterior, el dato de "Fuel Rate" no será medido en la fragata puesto que el algoritmo empleado no es de aplicación todavía al resto de unidades con SICP, pero sí se ha hecho en las lanchas de instrucción por su relación con el impacto ambiental, detallado en mayor profundidad en el Anexo I: Implicaciones Sociales, Económicas y Ambientales.

3.5.2 Modelo inicial en base a buques de guerra sensorizados

El programa ATAVIA, explicado en detalle en el apartado 2.3.1, cuenta en sus bases de datos con un total de 36 buques en constante monitorización. Es por ello, que se ha elegido esta plataforma como fuente de información para la obtención de los datos primarios con los que se generará las estructuras de ML de la aplicación.

Dentro de todas las unidades posibles, la fragata F-105 "Cristóbal Colón" es la más moderna y cuenta con el catálogo más completo de sensores y variables medidas en la base de datos a bordo. Esta es la plataforma de la que se han obtenido los datos, teniendo en cuenta las vicisitudes de un buque con propulsión CODOG.

Como se explica al final del apartado 3.5.1, la variable de consumo de combustible no ha sido elegida porque el algoritmo empleado a bordo no es representativo de los datos reales. Se calcula el consumo

medio en un periodo elegido por el personal operador de la consola del SICP, por lo que al no ser una medición periódica no se pueden obtener modelos representativos de su funcionamiento.

De acuerdo con el modelo MMG (2.6), las relaciones entre dinámica y revoluciones del buque pueden ser aproximadas por ecuaciones polinómicas, por lo que se ha optado por la aplicación de un modelo de este tipo para realizar las predicciones. Es un procedimiento que ajusta el modelo solicitado a una ecuación de la forma de la ecuación (16), siendo *n* el grado del polinomio. Las principales ventajas que ofrecen estos modelos son la flexibilidad y la simplificación de procesos por tratarse de un ajuste lineal.

$$y = \sum_{i=1}^{n+1} p_i x^{n+1-i} \qquad (16)$$

El detalle de las técnicas y código de Python empleados se encuentra en el apartado 3.5.4.

3.5.3 Toma de datos en LLII y tratamiento

Durante la ventana horaria que ofrecen las actividades de adiestramiento a flote, se han programado dos embarques en la lancha A-121 "GM Barrutia". El fin es la toma de datos para el modelado de la lancha, por medio de la creación de una hoja Excel equivalente a la de los buques del programa ATAVIA.

Al no disponer la ENM del material específico de CAT para la toma de datos, esta se realiza por medio de tomas de vídeo sincronizadas de los dos MPD de la cabina del puente. La medición se realiza bajo condiciones controladas, tomando el control de la plataforma el autor para navegar conforme a los siguientes requisitos resumidos:

ORDEN	DURACIÓN
AV. 600 RPM BR/ER	1 min
AV. 700 RPM BR/ER	1 min
AV. 800 RPM BR/ER	1 min
AV. 900 RPM BR/ER	1 min
AV. 1000 RPM BR/ER	1 min
AV. 1200 RPM BR/ER	1 min
AV. 1500 RPM BR/ER	1 min
REPOSICIONAMIENTO	

Tabla 3-4. Secuencia de órdenes a bordo de la lancha.
Elaboración propia recogida en el Anexo XII

Se ordenan dos rumbos, uno en dirección a la salida de la ría de Pontevedra, de componente suroeste, con el viento entrando por la popa, y otro de arribada a la Escuela Naval, de componente opuesta y viento por proa, de forma que se contemplen las situaciones favorables y desfavorables en el modelo.

La información completa de la prueba se encuentra en el Anexo XII: Documentación operativa para la toma de datos a bordo de las LLII, con la documentación operativa elaborada. Este documento sigue el modelo típico empleado por distintos astilleros que han construido buques para la Armada. En el protocolo se especifica el personal que participa en su redacción y en su ejecución. Anexados a este protocolo se encuentran también los resultados de la medición previos a los tratamientos realizados.

Al ser una toma de datos limitada, y por haber decidido emplear modelos de regresión lineal, se ha decidido ampliar la muestra por medio de la interpolación entre valores, aumentando la frecuencia de muestreo con una reducción de periodo de 15 a 3 segundos. Para ello, se ha empleado la herramienta Visual Basic de MS Excel, mediante una macro cuyo código se refleja en el Script 1 en lenguaje VBA.

```
Sub MacroDATAFILL()
Dim i, n As Integer
i = 1
n = 0

Sheets("Hoja1").Activate
Do While Range("A1").Offset(i, 0).Value <> ""
    Range("A1").Offset(i, 0).Resize(1, 16).Copy
    Sheets("Hoja2").Activate
    Range("A1").Offset(1 + n * 6, 0).PasteSpecial xlPasteValues
    Range("T3:AI7").Copy
    Sheets("Hoja2").Range("A1").Offset(2 + n * 6, 0).PasteSpecial
    Sheets("Hoja1").Activate
    n = n + 1
    i = i + 1
Loop
End Sub
```

Script 1

En el Script anterior, la matriz T3:AI7 hace referencia a una matriz incorporada en la misma hoja con las fórmulas correspondientes a una interpolación lineal entre los valores que se encuentren en la fila superior y la inferior, obteniendo cinco valores intermedios que aumentarán la frecuencia de muestreo de una forma artificial en los casos en los que los datos sean reducidos. Los resultados de la incorporación de este código se muestran para el caso de revoluciones del motor de babor (rpm) frente a velocidad de la plataforma (nudos) en la Figura 3-5. En ella, se puede observar cómo las mediciones se han realizado en momentos discretos, con el régimen de revoluciones ordenado en el procedimiento de la Tabla 3-4, en múltiplos de 100 revoluciones. Con este pretratamiento, se consigue obtener una visión de la trayectoria más definida entre las mediciones, útil para la selección del grado del modelo.

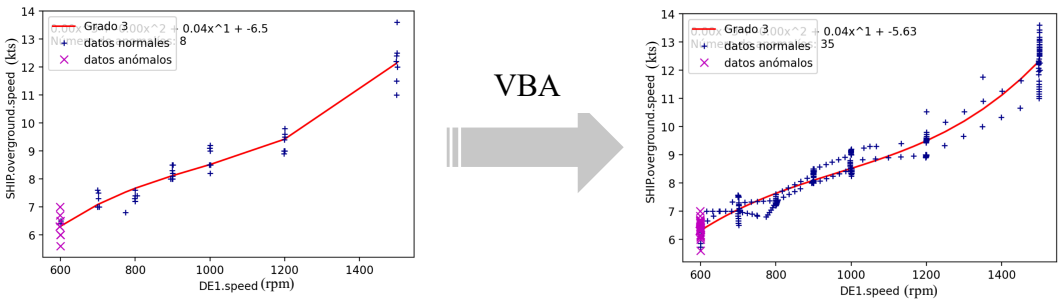


Figura 3-5. Resultados de la ampliación de datos con VBA.
Elaboración propia

3.5.4 Análisis de los datos

La operación de los buques está ligada a los parámetros observados en los ordenadores de los sistemas a bordo. Como se explica en el apartado 3.5.1, se han tomado los datos relacionados con la

operación de la propulsión, que se podrían definir como un conjunto de variables dependientes de las revoluciones del motor.

La librería de Python empleada para el análisis numérico es la de Scikit-Learn, importando las herramientas que se muestran en el Script 2.

```
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import PolynomialFeatures
from sklearn.metrics import r2_score
from sklearn import svm
```

Script 2

Los datos son filtrados conforme a lo establecido en la Zona "A" de la Página Data (Figura 3-9), por la fecha de la toma de datos y, en el caso de velocidades y revoluciones, los márgenes operativos de la plataforma. El desarrollo del código empleado para el el proceso de ML se divide en cinco etapas:

1. Selección de los datos de entrenamiento: se divide el contenido del dataframe correspondiente a las variables independiente (x) y dependiente (y) para que el 90 % de los datos sirvan para el entrenamiento del modelo y, el 10 % restante, para su validación. Esto queda establecido en la orden de la línea 2 del Script 3 con el comando *test_size*. Con la orden *values* de las líneas 3 y 4 de este Script se obtienen las matrices con los valores de cada *dataframe* y, con *reshape*, las matrices aseguran que su dimensión es de una columna y tantas filas como parámetros tiene el conjunto original.

```
# Selección de los datos de entrenamiento y de prueba, un 10 % del total
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.1)
X_train = X_train.values.reshape([X_train.values.shape[0], 1])
X_test = X_test.values.reshape([X_test.values.shape[0], 1])
```

Script 3

2. Nombramiento del tipo de algoritmo a emplear: con el Script 4, se crean las características polinomiales con las que se va a tratar el modelo. Estas características consisten en convertir los datos de un polinomio de grado i (parámetro *degree*) a otro de grado superior. El proceso que se sigue es la creación de tantas características como grado tendrá el polinomio, siendo estas de la forma que se muestra en la ecuación (17). Además, se ha optado por no incluir un sesgo constante en cada característica por estar construyendo los modelos sin una línea base conocida.

$$x_0 \Longrightarrow x^i + x \cdot x^{n-1} + \ldots\ldots + x \cdot x \qquad (17)$$

En función del grado seleccionado al inicio del proceso, deberán redimensionarse los valores de entrenamiento y prueba (*X_train* y *X_test*) a las matrices compatibles con el algoritmo, mediante la orden *arg.fit_transform*.

```
# Orden de cálculo del modelo polinomial de grado i
poly_features = PolynomialFeatures(degree=i, include_bias=False)
X_poly = poly_features.fit_transform(X_train)
Xt_poly = poly_features.fit_transform(X_test)
```

Script 4

3. Se emplea el algoritmo de regresión lineal de Sklearn en la línea 1 del Script 5, ajustándolo a continuación a las características polinómicas obtenidas anteriormente. Esta es la generación del modelo. Se definen en la línea 3 y 4 la forma de predecir la variable dependiente para los conjuntos de entrenamiento y de prueba respectivamente. Por último, se crean dos conjuntos de datos, las predicciones para cada variable de entrenamiento y para cada una de las de prueba, de forma que sea posible su posterior representación.

```
1   poly_model = LinearRegression()
    poly_model.fit(X_poly, y_train)
    pred = poly_model.predict(X_poly)
    pred2 = poly_model.predict(Xt_poly)
5   xt, yt = zip(*sorted(zip(X_test, pred2)))
    X, y = zip(*sorted(zip(X_train, pred)))
```

Script 5

4. Se ha añadido a partir de este punto un método de evaluación, el del coeficiente de determinación R^2 en el Script 6. Se calcula un valor para los datos de entrenamiento, y otro distinto para los datos de prueba (*r2_train* y *r2_test* respectivamente). Se basa en el empleo de la ecuación (18).

$$R^2 = 1 - \frac{\sum(Y - Y_{pred})^2}{\sum(Y - \bar{Y})^2} \qquad (18)$$

```
1   r2_train = r2_score(y_train, pred)
    r2_test = r2_score(y_test, pred2)
```

Script 6

5. Los datos se clasifican a continuación entre normales y anómalos mediante un algoritmo clasificador (*clf*) basado en la técnica *Support Vector Machine*, por medio de la clase *One Class SVM*, que permite trabajar con datos no etiquetados (aprendizaje no supervisado). Dado el parámetro *nu*=10 %, este porcentaje de los datos serán considerados anómalos. El parámetro *kernel* define el tipo de transformación que sufrirán los valores, siendo polinómica de grado *i*.
   Para realizar esta clasificación, se está generando un nuevo modelo con el ajuste de los datos de entrenamiento con SVM, expresado en la línea 2 del Script 7.
   La línea 4 indica cuales de los datos son considerados como anómalos, siendo aquellos que el algoritmo etiqueta con el valor *-1*. Los datos atípicos y normales se dividen en función de su etiqueta, siendo esta *[anomalias]* en el caso de los valores alejados o, en caso de los normales, *[~anomalias]*.

```
1    clf = svm.OneClassSVM(nu=0.1, kernel="poly", degree=i)
     clf.fit(X_train)


     anomalias = clf.predict(X_train) == -1
5    x_anomalos = X_train[anomalias]
     y_anomalos = y_train[anomalias]
     x_normales = X_train[~anomalias]
     y_normales = y_train[~anomalias]
```

Script 7

Continuando con los posibles análisis en la página "Data", la columna 2 de la Zona "A" (véase Figura 3-9) muestra la opción de realizar un filtrado. Estos filtros se han diseñado en base a una técnica de detección de datos anómalos *(outliers)*, mediante la aplicación de clasificación por cuartiles. Con el empleo de las *checkboxes* descritas en el apartado 3.6.1.2 es posible elegir en qué cuartil realizar el filtrado con respecto a la variable independiente. En cada uno de ellos, el proceso es el mismo que el que se describe en el Script 8, caso específico del primer cuartil.

La secuencia depende de la selección o no de la caja del filtro. En caso de selección, se filtran los datos del eje de abscisas (variable independiente), eligiendo los valores que cumplen los requisitos del cuartil correspondiente (datos mayores que el cuartil anterior y menores que el valor del cuartil seleccionado *Qxi*) en las líneas 9 y 10.

Dentro de esa preselección, se realiza una nueva clasificación, del primer y tercer cuartiles de la variable dependiente dentro de ese rango. Los datos elegidos finalmente serán aquellos que se encuentren dentro del rango de la fórmula (19):

$$Qy\_x11 - A \cdot IQR \leq Rango \leq Qy\_x13 + A \cdot IQR \qquad (19)$$

En esta ecuación, *Qy_x11* y *Qy_x13* hacen referencia a los valores del primer y tercer cuartil respectivamente de la variable dependiente *"y"* en el conjunto del primer cuartil de la variable dependiente *"x"*. El parámetro IQR es el rango intercuartílico, o la diferencia entre tercer y primer cuartil de datos, y el valor de "*A*" queda definido por el slider de la línea 6 del Script llamado "*Permisividad*", perteneciendo al conjunto [0;1,5]=$\{x \in \mathbb{R}: 0 \leq x \leq 1{,}5\}$. Como ejemplo, en el caso de que el valor de A sea nulo, el rango englobará aquellos valores que se encuentren dentro del rango intercuartílico.

```
Qx1=xper.quantile(0.25)
Qx2=xper.quantile(0.5)
Qx3=xper.quantile(0.75)
Qx4=xper.quantile(1)

filtro=st.slider(label="Permisividad",min_value=0.0,max_value=1.5,value=1.5)

if filtro1: #Siendo filtro1 una variable del tipo st.checkbox
    Qx1_valores = df.loc[xper < Qx1, nombre_varindep]
    y_x1 = df.loc[xper < Qx1, nombre_vardep]
    #Selección de los datos considerados como ruido
    Qy_x11=y_x1.quantile(0.25)
    Qy_x13=y_x1.quantile(0.75)

    #Rango intercuartílico
    IQR=Qy_x13-Qy_x11

    #Cálculo de los bigotes inferior y superior del box-plot
    BI1=Qy_x11 - filtro*IQR
    BS1=Qy_x13 + filtro*IQR

    nuevo_dfx1 = df.loc[(BI1 <= yper) & (yper <= BS1),nombre_varindep]
    nuevo_dfy1 = df.loc[(BI1 <= yper) & (yper <= BS1),nombre_vardep]
    else:
    nuevo_dfx1 = df.loc[xper <= Qx1, nombre_varindep]
    nuevo_dfy1 = df.loc[xper <= Qx1, nombre_vardep]
```

Script 8

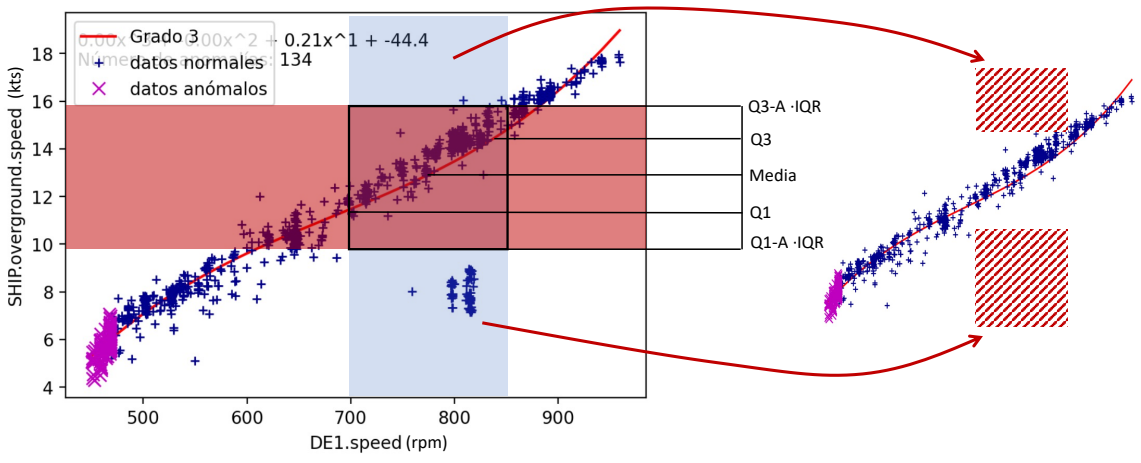


Figura 3-6. Representación del filtrado por cuartiles.
Elaboración Propia

3.6 Desarrollo de ventana de visualización del programa con Python

En esta sección se explica el desarrollo de la aplicación desde el punto de vista del diseño, en una parte más visual, así como particularidades funcionales (3.6.2) por medio de técnicas de eficiencia.

3.6.1 Streamlit

La aplicación muestra una interfaz desarrollada en su totalidad con la librería Streamlit. De lo general a lo particular, el diseño es similar al de una página web con 4 páginas integradas, seleccionables desde un menú en el lateral izquierdo. Para la construcción del menú, se debe crear un fichero tipo "workspace" en VS Code. Una vez creado, desde la IDE se añadirán empleando el menú lateral izquierdo en el apartado del explorador los archivos que se integrarán en la aplicación.

La aplicación se compone de 4 páginas, (archivos formato .py) y un documento config.toml. La configuración establece los colores a presentar en el texto y elementos de la librería, siendo estos los asociados a los códigos RGB[7] #173763 y #C7C7C7, ambos recogidos en el manual de imagen corporativa del AJEMA.

Los elementos comunes empleados en todas las páginas son:

- Cuadros de texto: se programan con la orden st.write("argumentos") introduciendo como argumentos el texto a mostrar.
- Títulos y subtítulos: análogos a los cuadros de texto mediante las fórmulas st.header("argumentos") y st.subheader("argumentos").
- Librería pillow (PIL) para el manejo de imágenes (escudos de la Armada y Escuela Naval en las barras laterales).

3.6.1.1 Página Home

La página Home hace uso de las librerías PIL y Pandas además de Streamlit. Al abrirla, se le pide al usuario introducir el nombre del archivo contenedor de los datos, en formato MS Excel, mediante el Script 9. Si se ha elegido un archivo, comprobado por medio de un comando *if "arg" is not None,* el programa almacenará el libro en la variable *xls* y leerá los nombres de las hojas que contiene. Aparecerá entonces un menú desplegable que muestra la opción de elegir la hoja que contiene los registros de datos, con el comando de Streamlit *st.multiselect* con una selección admitida como máximo. De nuevo, por medio de otro *if*, si se selecciona la hoja, automáticamente se almacenarán los datos que contiene en el *dataframe* llamado *df.*

[7] Se trata de un espacio de color que pretende reproducir los colores del espectro visible gracias a la mezcla de los colores rojo (Red), verde (Green) y azul (Blue), dando lugar a lo que se conoce como síntesis aditiva del color. Se expresa en base hexadecimal un número de 8 bits por cada color (rango entre 0 y 255) para la composición de la mezcla, y su código se corresponde con la unión de los 6 valores en orden R-G-B.

```
excel_file = st.file_uploader("Selecciona un archivo Excel", type=["xlsx", "xls","xlsm"])
# Si se seleccionó un archivo, guardamos su ruta
if excel_file is not None:
    xls = pd.ExcelFile(excel_file)
    #Con la librería pandas se obtienen los datos de las hojas que contiene el Excel
    sheet_names = xls.sheet_names
    # Adición de un multiselect para seleccionar la hoja que contiene los datos de la plataforma
    selected_sheets = st.multiselect("Seleccione la hoja a analizar", sheet_names,max_selections=1)
    if len(selected_sheets)==1:
        # Cargar los datos del archivo de Excel
        excel_file = f'{excel_file.name}'
        data_name = selected_sheets[0]
        df = pd.read_excel(excel_file, sheet_name=data_name, usecols='A:O', header=0)
```

Script 9

A continuación, se realiza automáticamente una correlación de variables empleando el método de Spearman, explicado en el apartado 2.9.1.2. Muestra la información básica de empleo de la aplicación por medio de un cuadro de texto. El código completo empleado en la página se muestra en el Anexo VI: Código Fuente Página Home.

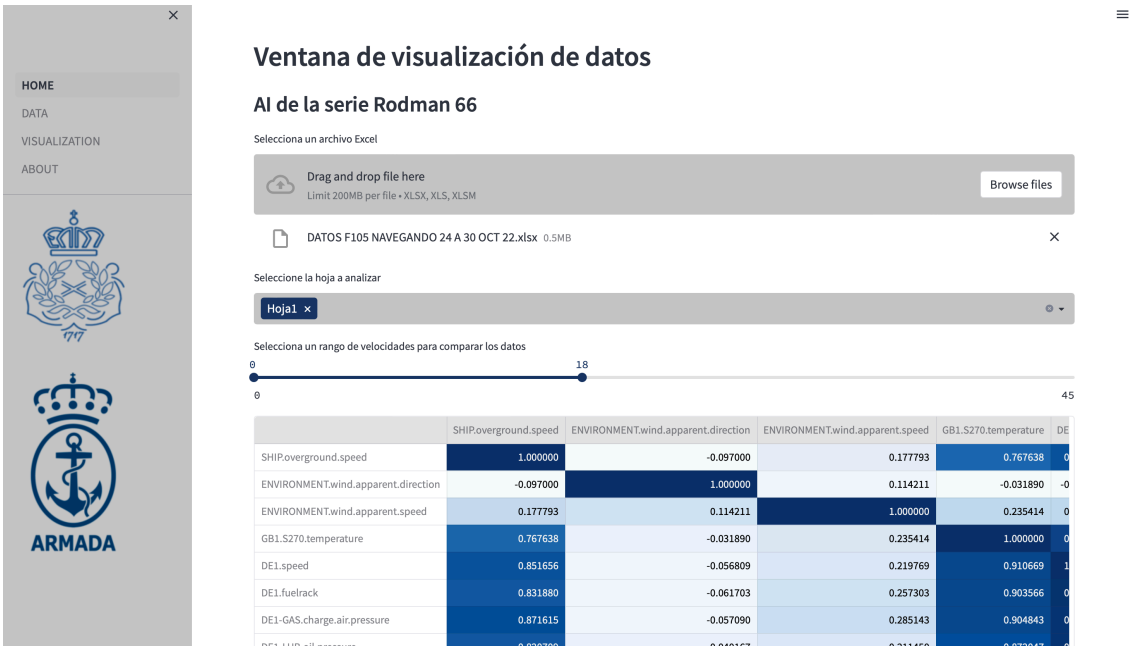


Figura 3-7. Página "Home"
Elaboración propia

Para el cálculo de los coeficientes de Spearman, se ha empleado la librería PANDAS, con la función arg.*corr(method="spearman")*, que calcula los coeficientes de correlación de los argumentos *"arg"* de acuerdo con el método elegido. Las opciones para representar son los coeficientes de Pearson, Spearman y Kendall. Se han elegido los coeficientes de Spearman por las siguientes razones:

- Magnitudes adimensionales: los cálculos se realizan con respecto a la correlación de rangos entre ambas variables, entendiéndose como tal al lugar que ocupa en el conjunto total de valores, como se explica en el apartado 2.9.1.2. De esta forma, la relación que representa el coeficiente no está sujeta a cambios de gradiente, curvas y otras trayectorias, como sí lo hacen en el coeficiente de Pearson.
- Resistencia a datos atípicos: el tratamiento de los datos, explicado en el apartado 3.5.4, muestra la existencia de datos atípicos en las observaciones, ocasionados por errores de medida, u otros fallos en la obtención de los parámetros observados. Frente a la razón de trabajar con magnitudes adimensionales se plantea la posibilidad de elegir el modelo de Pearson o el de la Tau de Kendall, que sigue la ecuación (20)[8]. Kendall ofrece una mayor fidelidad al grado de correlación, siendo una opción mejor para conjuntos con muestras pequeñas ($n < 12$), aunque es una medición afectada por valores atípicos en mayor medida que la opción de Spearman. Además, la función que debe representar este análisis en la página de "home" es la de mostrar al personal investigador las correlaciones que existen entre variables, para elegir entre cuales de ellas tendría valor estudiar los modelos. La escala de color en tonos azules (de la escala *cmap='Blues'*) incrementa la intensidad del color, y por tanto la atención, en valores próximos a los extremos de las funciones, ± 1. Con el cálculo de la rho de Spearman, el renunciar a la precisión hace que los valores de los coeficientes sean más extremos, por lo que favorece a esta dispersión y, por lo tanto, a la labor visual, como se muestra en la Figura 3-8 [46].

$$T_{XY} = \frac{2 \cdot (P - M)}{n(n-1)} \qquad (20)$$

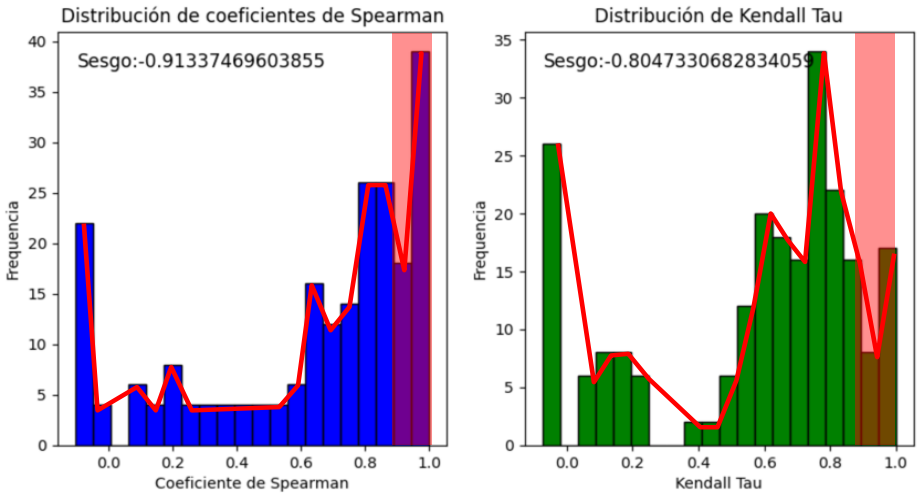


Figura 3-8. Distribución de los coeficientes de correlación
Spearman frente a Kendall
Elaboración propia

De una forma numérica, se han calculado los sesgos de ambas distribuciones con la función *skew* de scikit learn como se muestra en las etiquetas superiores izquierdas de la Figura 3-8, siendo más negativo

[8]En la ecuación, P se corresponde con el número de acuerdos, es decir, la cantidad de veces que con un incremento de la variable X se produce un incremento de la variable Y, o el número de veces que $y_i < y_j$ para todo $i < j$. El valor de M, por el contrario, se corresponde con el número de desacuerdos, la cantidad de veces que $y_i > y_j$.

el caso de los coeficientes de Spearman. La interpretación de estos datos demuestra que los valores se distribuyen con un sesgo más pronunciado a la derecha, próximo a los valores máximos positivos de los coeficientes (sombreado en color rojo).

Cabe destacar que, en la figura, los datos introducidos para el cálculo de coeficientes son de la fragata "Cristóbal Colón" y han sido filtrados entre aquellos en los que la velocidad se encontrase entre los 0 y 18 nudos por los motivos expuestos más adelante.

La representación final parte del código que se muestra en el Script 10. En él, la ruta al archivo seleccionado ("*excel_file*") se obtiene con la orden *.name*, a partir de la cual puede leer el contenido del documento y proceder a su tratamiento.

Antes de mostrar los coeficientes de Spearman, aparece un filtro que permite elegir la velocidad mínima y máximas a observar, para que en casos como las fragatas F-100 no haya conjuntos con datos de propulsión a turbina mezclados con propulsión diésel.

```
#Cargar los datos del libro de Excel, almacenado como primer elemento del
array selected_sheets
excel_file = f'{excel_file.name}'
data_name = selected_sheets[0]
df = pd.read_excel(excel_file, sheet_name=data_name, usecols='A:R',header=0)
        vel_inicial=0
        vel_final=45


#Crear slider de doble fin limitado por las velocidades a analizar. Se
elegirán velocidades entre los 0 y 45 nudos, este último por ser la máxima
velocidad alcanzada en un buque de guerra en la historia (Destructor francés
"Le Terrible, 1935")
minv_elec, maxv_elec = st.slider("Seleccione un rango de velocidades para
comparar los datos",
    value=(vel_inicial, vel_final),
    min_value=vel_inicial,
    max_value=vel_final)


speed=df['SHIP.overground.speed']
df = df.loc[(speed<=maxv_elec) & (speed>=minv_elec),:]


spearman_corr = df.corr(method='spearman')
st.write(spearman_corr.style.background_gradient(cmap='Blues'))
```

Script 10

3.6.1.2 Página Data

En esta página se muestran las opciones de análisis de datos de la plataforma, es la más variada en contenido de la librería Streamlit, así como de otras que han servido para trabajar de una forma más visual.

La configuración inicial de la página incluye su título y subtítulo, de la misma forma que en el caso anterior. A continuación, se solicita la selección de un archivo, petición realizada por medio de los mismos comandos que el Script 9, explicados en la Página Home.

A continuación de la selección de todos los datos y su almacenamiento en una variable (*df*), se pide la selección de una variable dependiente y otra independiente, que serán confrontadas y con las que se podrán crear los modelos como se explica en el apartado 3.5.4. El contenido del *dataframe* puede ser modificado mediante los filtros de fechas visualizar con otro elemento st.slider y mediante los parámetros operativos modificados por la técnica Poka Yoke (velocidad y revoluciones).

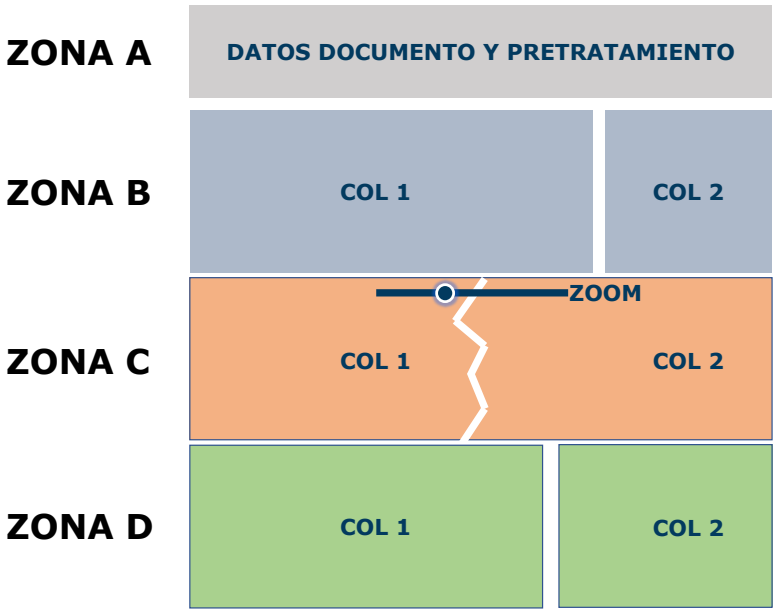


Figura 3-9. Esquema de diseño de la página "Data"
Elaboración propia

El diseño de la página a partir de este momento se divide en columnas con unas relaciones de anchura que podrán ser fijas o variables, como se muestra en la Figura 3-9. Se explican a continuación los contenidos de cada una de las zonas representadas:

- Zona "A": muestra las herramientas de selección de documentos y filtros básicos explicados anteriormente. Una única columna para toda la información (expresado en el código como *use_column_width=True*). También contiene un widget *st.number_input* que permite seleccionar el grado del modelo a emplear, un número entero entre 1 y 6. Se ha elegido 6 como grado máximo porque, de los modelos representados, todos eran válidos con polinomios de grados inferiores a 5, siendo las opciones siguientes las asociadas a modelos sobreajustados. El valor que devuelte este elemento es un valor de tipo entero (*int*) almacenado en la variable *i*, como se puede ver en la Figura 3-10.



Figura 3-10. Zona "A".
Elaboración propia

- Zona “B”: es un área destinada a la visualización de los datos que han sido seleccionados en la zona “A”. Está dividido en dos columnas con anchos prefijados, siendo la columna número 1 el doble de ancha que la número 2.
    - Dentro de la columna número 1 (COL 1) se representa un gráfico de dispersión con todos los datos que cumplen las condiciones de los filtros previos, así como el modelo de regresión polinómica de grado *i* que el código de Scikit ha sido capaz de determinar. Dentro del área del gráfico se muestran la ecuación del modelo determinado y el número de valores atípicos detectados con el algoritmo SVM por ser el 10 % de datos más alejados del modelo, como se explica en el apartado “Análisis de los datos”.
    - En la columna número 2 (COL 2) se muestran cuatro casillas o *checkboxes* que permiten aplicar un filtro estadístico basado en el reparto de los datos en cuartiles, explicado en el apartado 3.5.4. Ese filtrado se puede ver en otra representación más pequeña de un gráfico de dispersión de las mismas características que el de la columna 1. Para este menor número de datos, se vuelve a calcular un modelo más ajustado y se obtiene la misma información que en la columna anterior.

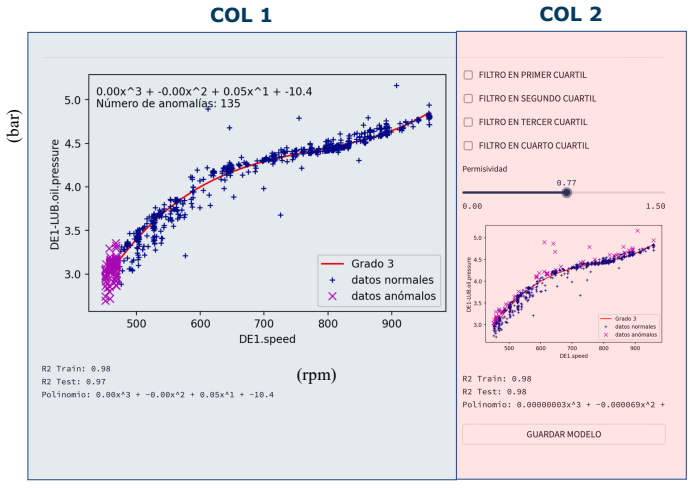


Figura 3-11. Zona "B".
Elaboración propia

- Zona “C”: se trata de un espacio de dos columnas de dimensiones personalizables. El widget *st.slider* de *Streamlit* ha sido configurado para que, con su movimiento a izquierdas o derechas, varíe el ancho de las columnas de cada uno de los lados, de forma que sea posible ampliar los gráficos que contiene.
    - En la columna número 1 (COL 1) se representa un histograma de los datos iniciales de la variable dependiente, con las frecuencias de repetición en el eje de ordenadas y los valores, en el de abscisas.
    - En la columna número 2 (COL 2) se representa un diagrama de caja y bigotes (*box plot*) con los mismos datos que el histograma anterior. Esto permite también ver, de forma más exacta, el contenido del conjunto inicial de datos.

  En ambas columnas, los gráficos han sido representados con la librería *Plotly Express* para que permitan una mayor interacción mostrando los detalles de cada gráfico al deslizar el cursor sobre ellos. En la representación de la columna 2 en la Figura 3-12 se puede ver el efecto producido al posicionarse sobre el diagrama de caja y bigotes.

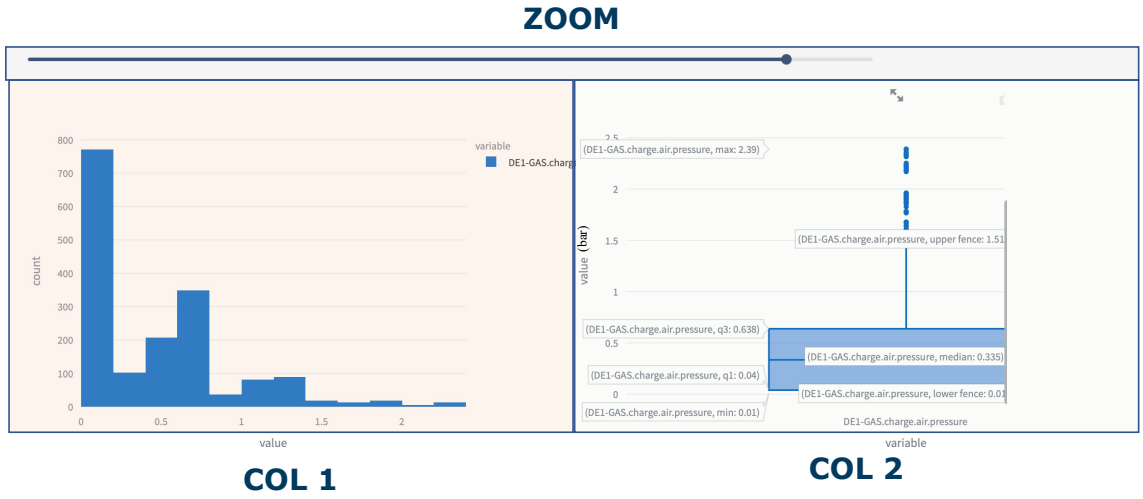


Figura 3-12. Zona "C".
Elaboración propia

- Zona “D”: en este último área, se divide el espacio en dos columnas de tamaños fijos. Se muestra el modelo generado a partir del conjunto de datos filtrados por cuartiles (Zona “B”,COL 2), con su representación gráfica y un comprobador numérico.
    - En la columna número 1 (COL 1) se representa el modelo del grado elegido en el selector *st.number_imput* de la zona “A” ajustado para los datos filtrados en por cuartiles en la zona “B” columna 2. Además, con la librería *Seaborn* (*sns*) se hace uso del comando del Script 11 para representar un intervalo de confianza a lo largo del gráfico del 95 % con los datos filtrados de la variable independiente y dependiente, *nuevofiltrox* y *nuevofiltroy* respectivamente. En el gráfico se representan también los valores que hayan sido introducidos en la columna 2, sean unas coordenadas dadas (comprobador de anomalías) o un punto del modelo con su variable dependiente obtenida con una predicción.

```
1    sns.regplot(x=nuevofiltrox,y=nuevofiltroy, ci=95,
     scatter=False,order=i,line_kws={"alpha":0},label='Distribución de datos')
```

Script 11

    - En la columna número 2 (COL 2) es posible hace uso del comprobador de anomalías y de la herramienta de predicción. El comprobador permite la introducción de valores para variable dependiente e independiente, que son graficados en en el

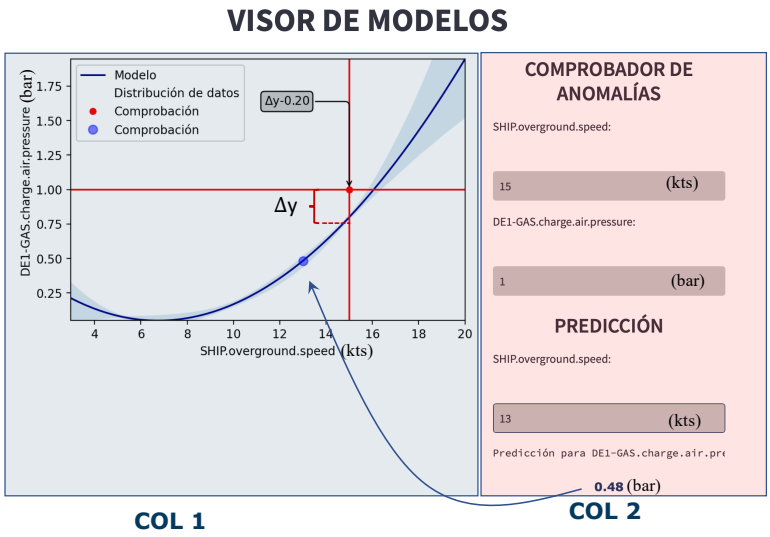


Figura 3-13. Zona "D".
Elaboración propia

En general, esta página es la que contiene el peso del análisis de datos, y cuenta para ello con herramientas que permitan visualizar los resultados obtenidos. La Figura 3-14 muestra la página en su totalidad.

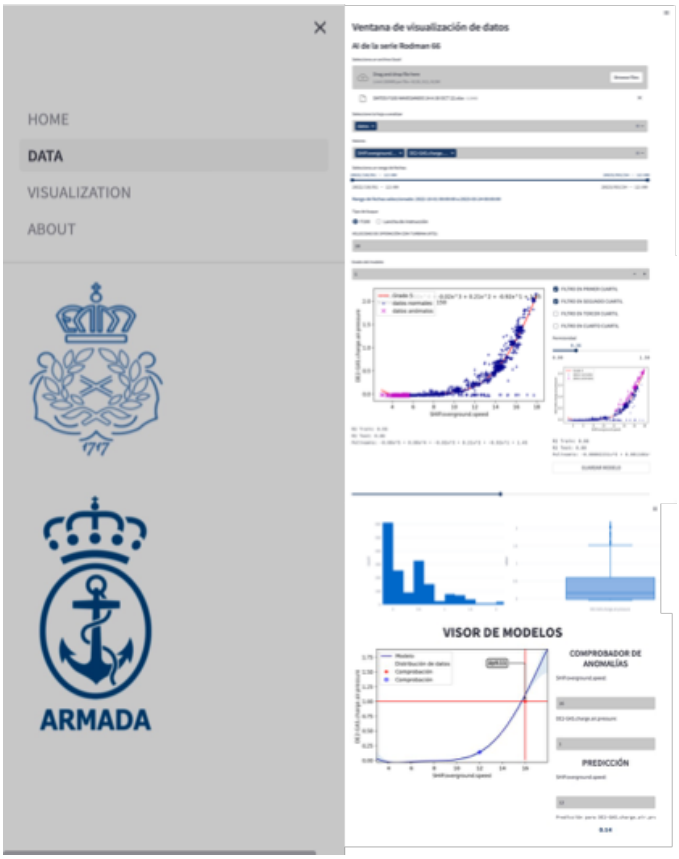


Figura 3-14. Página "Data" vista general.
Elaboración propia

3.6.1.3 Página Visualización

Esta página contiene el enlace para la ejecución del programa desarrollado por Unreal Engine (más información en el apartado 3.8), para la visualización del interior de la lancha en primera persona. Al

iniciarla, se muestra el título y un botón que, al presionarlo, lleva un enlace a través de la ruta del ejecutable. Su ejemplo en funcionamiento se muestra en la Figura 3-15. El código empleado en esta página se encuentra disponible en el Anexo VIII: Código Fuente Página Visualización.



Figura 3-15. Página "Visualization".
Elaboración propia

3.6.1.4 Página About

La página “sobre la aplicación” ha sido realizada para dar a conocer el origen de esta, sin más complejidad que el título y un cuadro de texto para contener su enunciado. El código empleado en esta página se encuentra disponible en el Anexo IX: Código Fuente Página About.



Figura 3-16. Página "About".
Elaboración propia

3.6.2 Poka Yoke

Para llevar a cabo la tarea desarrollo, en cuanto a la interactuabilidad del usuario se ha empleado la técnica "Poka Yoke", o diseño a prueba de errores. Un ejemplo común de este método en el día a día es el diseño de un conector tipo USB, que únicamente puede ser introducido en el sentido adecuado.

De esta forma, en caso de que el usuario seleccione una variable que presente irregularidades en unidades, el modelo solicitará un valor de entrada antes de continuar con cualquier otra instrucción. Los valores que se han elegido son la velocidad y las revoluciones, puesto que los modelos trabajan sobre las variables de los motores diesel, pero en unidades como las fragatas clase Álvaro de Bazán este modo de operación se reduce a velocidades inferiores a los 18 nudos. Al darse la posibilidad de estar operando en modo turbina (propulsión Combined Diesel Or Gas/CODOG) pero aun así encontrarse por debajo de los 18 nudos por condiciones meteorológicas adversas, por ejemplo, si se seleccionan las revoluciones del motor también aparecerá una solicitud de la velocidad de ralentí de los motores (válido también para descartar valores atípicos en otras unidades).

En el caso de datos de la lancha de instrucción, no aparece la opción de operación con turbina

También, el programa general desarrolla un funcionamiento escalonado, manteniéndose a la espera de las entradas del usuario antes de ejecutar ninguna orden.



Figura 3-17. Solicitud de parámetros de la plataforma Poka Yoke en la ventana de visualización
Elaboración propia

3.7 Modelado 3D del interior de la LLII

Al ser uno de los objetivos del trabajo la creación de una visualización de los modelos, una posibilidad que replicaría los resultados de los gemelos digitales más empleados en el ámbito de defensa es la representación en un entorno tridimensional de los sistemas monitorizados. Esto implica una fase de modelado del sistema en un entorno 3D y otra de programación de la monitorización y presentación.

En un primer momento, se plantean dos opciones para realizar el modelado 3D:

- A partir de un software CAD diseño integral de la plataforma de acuerdo con lo establecido en los planos de Rodman. No sólo es necesario replicar estos planos, sino que también sería necesario una toma de medidas in situ de la propia lancha, ya que los equipos y sistemas no instalados por el astillero no están recogidos en dicha documentación. Los modelos generados mediante esta metodología cuentan con mayor precisión y fidelidad en la representación con la realidad, aunque el esfuerzo temporal que implica es mayor.

- A partir de técnicas de ingeniería inversa, replicar los sistemas para su posterior integración en la ventana de visualización. Esto implica el empleo de medios especializados y, como beneficios, ofrece resultados con un menor esfuerzo temporal que en la opción anterior.

Dado el alcance del trabajo, sus necesidades y los riesgos inherentes a cada opción, se hace uso de la Tabla 3-5 para la selección de una metodología a emplear, eligiendo las técnicas de ingeniería inversa como opción final.

	BAJO/NO	**MEDIO**	**ALTO/SI**
COSTE COMPUTACIONAL	-	DISEÑO CAD	ING. INVERSA
COSTE TEMPORAL	ING. INVERSA	-	DISEÑO CAD
CONOCIMIENTOS PREVIOS ADICIONALES REQUERIDOS	-	DISEÑO CAD	ING. INVERSA
NECESIDAD DE DOCUMENTACIÓN	ING. INVERSA	-	DISEÑO CAD

Tabla 3-5. Matriz de decisión con ejemplos para selección de metodología de modelado 3D.
Elaboración propia

3.7.1 Escaneo LIDAR y óptico

La ingeniería inversa, de una forma más detallada, consiste en la captación de la información geométrica de un objeto físico procesándola en un entorno tridimensional mediante software especializado. El replicado de estos elementos puede realizarse de diferentes formas, entre las que se incluyen [47]:

- Toma de mediciones con instrumentos debidamente calibrados.
- Escaneado LIDAR (Light Detection and Ranging).
- Palpado mecánico.
- Digitalización óptica por medio de fotogrametría.
- Tomografías axiales computarizadas (rayos X).

En un primer momento se opta por el escaneado LIDAR con el escáner Leica RTC 360, equipo proporcionado por el Centro Universitario de la Defensa. Se trata de una herramienta portátil capaz de llevar a cabo registros de imágenes de alto rango dinámico (HDR) y sistema inercial visual que permite llevarlos a cabo en tiempo real. Su rango de alcances abarca desde los 50 centímetros de mínimo hasta un máximo de 130 metros, y su sistema de captación de imágenes a todo color se compone de 3 cámaras de 36 MP capaz de capturar datos brutos de 432 MPx para una imagen esférica calibrada de 360° x 300°.

A bordo de la lancha A-121 “GM Barrutia” se llevó a cabo un montaje en tres estaciones diferentes en el local del puente (Figura 3-18) para obtener una nube de puntos con mayor precisión y eliminando las zonas de sombra generada por los equipos que se pudieran interponer entre el sensor y la superficie de la cabina.

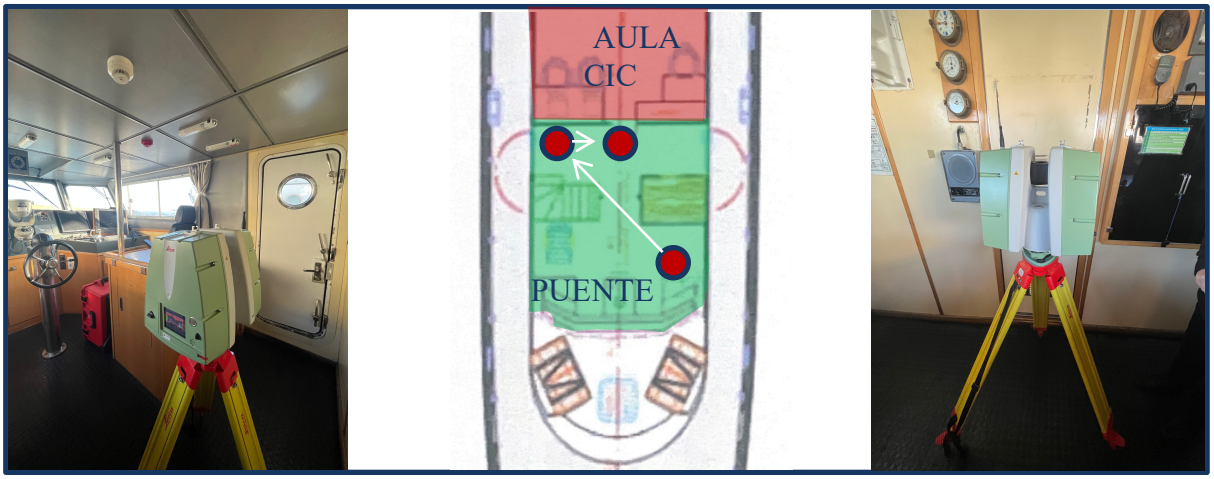


Figura 3-18. Esquema del emplazamiento del sensor Leica RTC 360 en el puente de la A-121.
Elaboración propia

El resultado del escaneo fue procesado en el software proporcionado por la casa del escáner, el Leica Cyclone Register 360, con el que se obtuvo una nube de puntos con resultados muy precisos en algunas regiones, aunque con algunos espacios sin información. El informe generado por el software se encuentra disponible en el Anexo V: Informe Leica Cyclone Register 360 PLUS. Los errores registrados se deben a la imposibilidad del programa de procesado de posicionar el tercer estacionamiento en su ubicación exacta, generando así una nube incompleta. Se intentó un segundo tratamiento eliminando la estación que generaba los fallos, dando lugar a mejores resultados, aunque con errores todavía, como se muestra en la Figura 3-19.



Figura 3-19. Resultados tratados por Cyclone 360 (izq) y exportados a Geomagic (der).
Elaboración propia

A la vista esta obtención, estos modelos deberían ser tratados con software adecuado para completar las nubes de puntos, un proceso que podría retrasar las tareas con dependencia de ésta.

Aplicando la metodología de gestión de riesgos explicada en el subapartado 3.4, se identifican los trabajos necesarios de postprocesado del escaneo como una fuente de riesgo. Los eventos o riesgos derivados que implicaría son los siguientes:

1. Imposibilidad de finalizar el mallado.
2. Retrasos en la fase de modelado
3. Retrasos por dependencias de otras etapas del desarrollo.
4. Generación de baja calidad del mallado, con baja resolución.
5. Generación de un nube de puntos que no permita que el sólido derivado sea adecuado para su empleo posterior.

Que en la matriz de probabilidad frente a impacto quedarían distribuidos de la forma siguiente:

Probabilidad \ Impacto	1	2	3	4	5
5	-	④	-	-	③
4	-	-	-	②	-
3	-	-	-	-	⑤
2	-	-	-	-	①
1	-	-	-	-	-

Tabla 3-6. Matriz de gestión de riesgos del procesado del escaneo de Leica
Elaboración propia

Las puntuaciones asignadas conforme a la matriz para cada riesgo son:

1. 50 puntos.
2. 64 puntos.
3. 125 puntos.
4. 16 puntos.
5. 75 puntos.

El total de puntos es de 330 puntos, valor que se ha considerado lo suficientemente elevado para buscar otras soluciones alternativas cuyo coste sea menor.

Dados los efectos negativos que podrían acarrear, se plantea realizar un segundo escaneo, únicamente por medio de técnicas de digitalización óptica selectivas, escaneando conjuntos que por su geometría y dimensiones no supongan un gran coste computacional y sean fácilmente integrables a posteriori, eliminando los riesgos representados en la Tabla 3-6. Se emplea para la toma de imágenes la cámara de un IPhone 13 y se dividen las mallas en los siguientes conjuntos:

- Sistemas del puente-proa: 1132 fotografías.
- Mamparo y elementos banda estribor: 39 fotografías.
- Mamparo y elementos banda babor:12 fotografías.
- Mamparo de popa: 17 fotografías.

El procesado de las imágenes se realiza en la aplicación 3D Scanner App, obteniendo como resultado los modelos en formato .obj de Wavefront, como se muestra en la Figura 3-20.

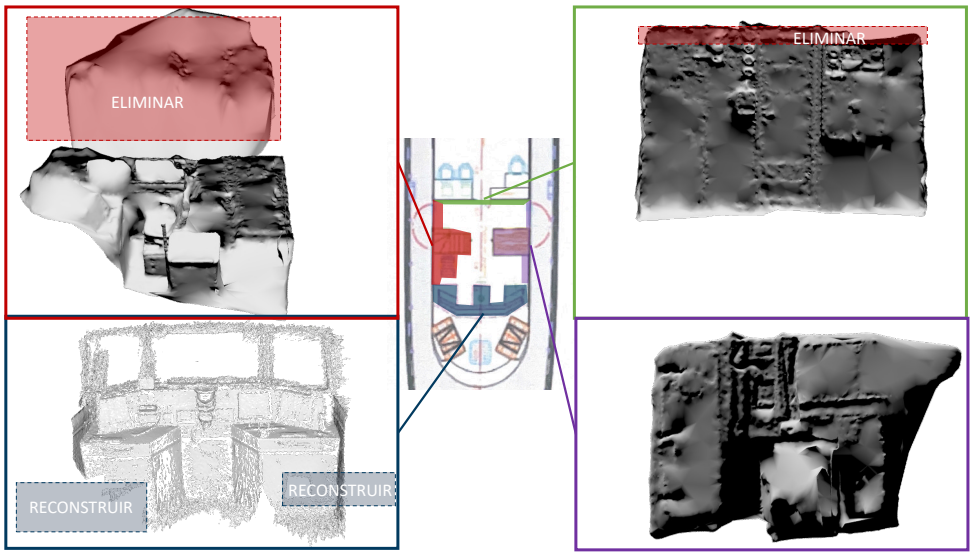


Figura 3-20. Mallas obtenidas con digitalización óptica.
Elaboración propia

Estos archivos continúan necesitando de un postprocesado, pero en una menor medida que en los realizados por el escáner de Leica, a pesar de tener menos resolución que este último. Las 4 mallas han sido exportadas a la aplicación Blender para su renderizado, eliminando así las nubes de puntos restantes, que acostumbran a aparecer en las intersecciones de los mamparos con el techo de la cabina y en la banda de estribor, en la que la ventana permite ver la lancha abarloada a una distancia que el procesador considera como parte del modelo 3D.

Para la eliminación, desde el modo edición en Blender, se hacen visibles los vértices que se han identificado y se suprimen con especial atención a las superficies que se ven afectadas. Este archivo es exportado a formato FBX para tener una mayor interoperabilidad entre los software de desarrollo, así como son importadas también sus texturas para replicar las imágenes captadas por la cámara.

Los 4 elementos son unidos en la fase de programación en un único sólido con el que se puede interactuar y con características propias

3.8 Entorno inmersivo con Unreal Engine 5

3.8.1 Generalidades

Los objetivos de visualización se pretenden alcanzar mediante el desarrollo de una plataforma similar a un juego en primera persona (FPV), por lo que el paquete inicial elegido desde Unreal Engine 5 será el First Person Character Project. La cámara desde la que se visualizará la interfaz estará enmarcada en esta visión en primera persona, con los controles preestablecidos por Unreal en los Blueprints del *First Person Character*. La orientación de la cámara se ordena con el movimiento del cursor y, el desplazamiento, mediante las teclas de acción “w”, “s”, “a” y “d” para adelante, atrás, izquierda y derecha respectivamente.

3.8.2 Creación del terreno

En la fase de formación, se comienza con los principios de empleo de esta herramienta, desarrollando escenarios de simulación que puedan ser empleados en este proyecto. El primer desarrollo consistía en un escenario simulando la ría de Pontevedra, base del actual proyecto, pero con escasa explotación todavía.

El mapa del terreno ha sido generado gracias a los conocidos como "heightmaps" o mapas de alturas, una imagen de una vista aérea del terreno. El editor de terreno, por el gradiente de colores en una escala en blanco, negro y grises es capaz de asociar los puntos de la imagen al desplazamiento de un plano. Para aplicar la textura del terreno, se toma una vista satélite de la misma zona limitada por el heightmap como se ve en la Figura 3-21, donde la curva de nivel naranja marca los puntos a la misma altura, o con el mismo color en la escala de grises. La línea roja marca el extremo del plano visto en las tres representaciones, abajo con la altura aplicada.

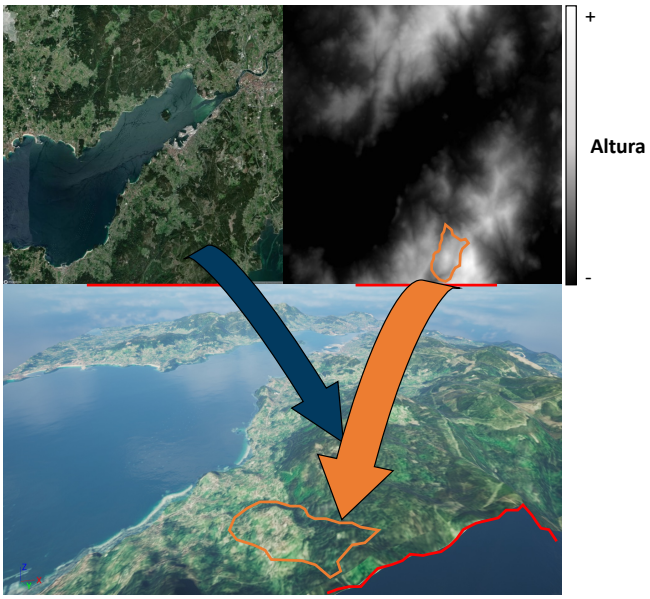


Figura 3-21. Elaboración del terreno en UE con heightmaps
Elaboración propia

Las texturas son elementos de Unreal Engine que definen la forma en la que se ve su contenido, pudiendo tratarse desde colores planos hasta complejas imágenes modificadas por medio de código multinodal, al que se referirán como Blueprints o BP. Por defecto, al crear una textura desde una imagen, el software creará un patrón de repetición para rellenar la superficie en la que se aplique, lo conocido como "tiling" o embaldosado por su equivalencia con un conjunto de baldosas. Será necesario para este caso eliminar toda multiplicación para asegurar que la imagen satélite no tiene réplicas. La Figura 3-22 muestra el Blueprint que se ha creado a partir de la imagen satélite, añadiendo el elemento Texture Coordinates enlazado a el canal UV[9] de la muestra, de forma que es posible modificar los parámetros de embaldosado en los ejes horizontal y vertical con UTiling y VTiling respectivamente.

El heightmap original empleado tiene una resolución de 1024x1024 pixeles, mientras que la imagen de la textura tiene 562x562 px. Además, el terreno se ha modificado para que su tamaño sea el equivalente al tamaño real del mapa, escalando los ejes del plano multiplicándolos por 100 en cada sentido de los ejes X e Y y por 10 en el eje Z (la altura se ve más influenciada por el valor del Alpha elegido en el desplazamiento). La reconversión necesaria será la división de las escalas en cada eje (U,V) entre el producto de la escala actual del mapa en ese eje, la escala en el eje vertical y la relación entre las imágenes originales.

[9] El canal UV es un conjunto de datos que relacionan los puntos de una malla estática tridimensional con las coordenadas de un plano en dos dimensiones. La U hace referencia a las coordenadas horizontales y la V, a las verticales.

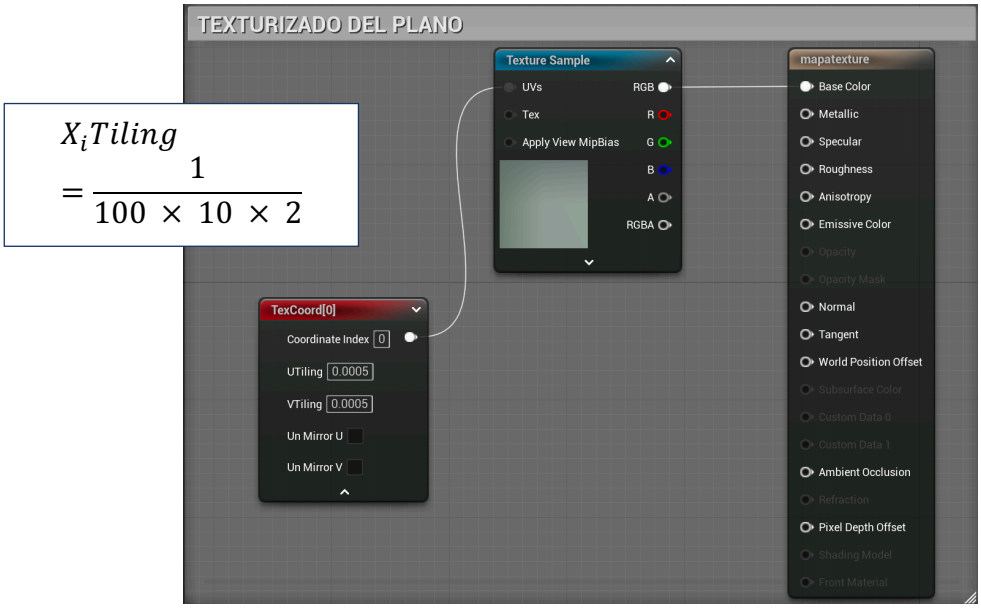


Figura 3-22. Blueprint textura terreno.
Elaboración propia

3.8.3 Importación de los modelo 3D

A partir de los modelos en 3D obtenidos en el escaneo (más información en el apartado 3.7.1), éstos son integrados en los elementos del simulador como mallas estáticas, importados desde formato .fbx. Para que en la visualización no sea posible atravesar los mamparos y sistemas del barco, es necesario establecer una malla rígida, que cuente con características de colisión. En un primer lugar, se desarrolla una base en el programa SketchUp de Trimble, generando un modelo adicional, con forma de paralelepípedo y las medidas de la lancha como un prisma de base rectangular. Se incluyen en él los modelos simplificados de las estructuras principales de la cabina, como los paneles en proa y las dos mesas, asegurando así que, en caso de que los escaneos carezcan de puntos en los que establecer la malla de colisión, sí existirá una restricción por este modelo.

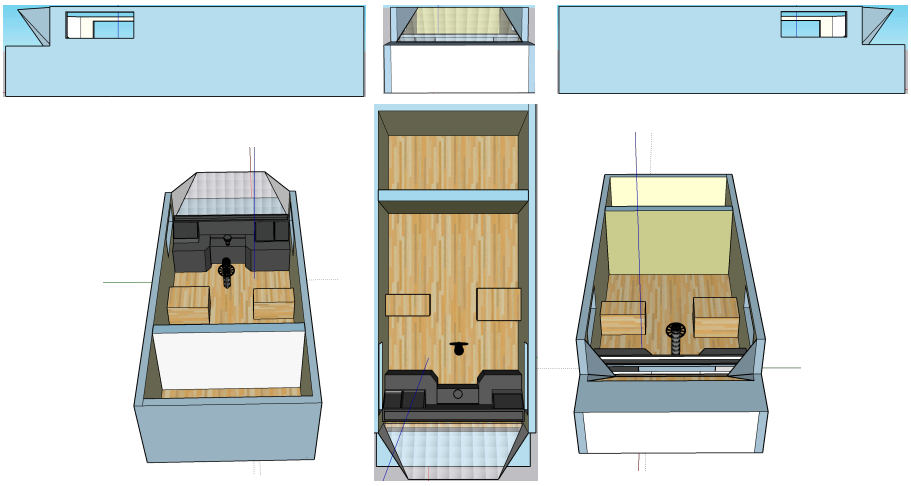


Figura 3-23. Modelo generado con SketchUp
Elaboración propia

Al importarlo al contenido del proyecto, para añadir características de colisión, se debe seleccionar la opción de editar el Blueprint de cada modelo y, desde el panel que aparece en el lateral derecho, seleccionar una complejidad de colisión del tipo “colisión compleja como simple”. Esto definirá la geometría de la malla que se genera alrededor del objeto 3D y que no se podrá traspasar una vez se

ejecute el programa. En el caso de usar una colisión simple, se generará un espacio paralelepípedo que contenga la totalidad del modelo en su interior, limitando en mayor medida el acceso a zonas del modelo.

En la Figura 3-24, la leyenda simple-complex muestra la vista en planta de la colisión generada para dos nubes de puntos iguales en función del algoritmo empleado, siendo más preciso el complejo. Para un espacio como el interior de las lanchas de instrucción, de reducidas dimensiones, es conveniente maximizar el espacio de movimiento para eliminar el posible efecto de errores residuales en la generación de las nubes de puntos.

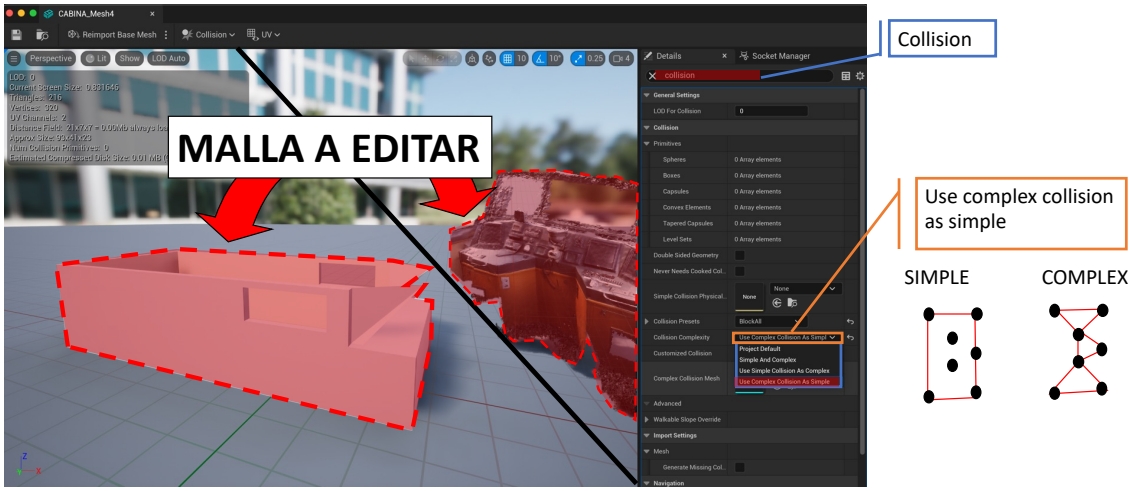


Figura 3-24. Explicación tipo de colisión empleada
Elaboración propia

3.8.4 Importación de datos de buques

El motor Unreal Engine es capaz de leer documentos en formato .csv, pudiendo presentar y manipular su contenido. Para ello, es necesario realizar un pretratamiento de los datos.

Desde UE, es necesario definir una estructura del *dataframe* a importar, el equivalente a establecer el "esqueleto" del archivo, los nombres de las columnas y el tipo de valor que contienen. Esto se realiza añadiendo dentro de la carpeta donde se quiera almacenar el archivo de datos una estructura, ubicada en la ruta *Add* → *Blueprints* → *Structure*. Dentro de este nuevo elemento, se añadirán tantas variables como columnas contiene el documento a importar, con el mismo nombre que figura en la cabecera de la tabla. Se deberá definir el formato de los datos que contiene entre el listado que que se muestra en la Figura 3-25. Para este trabajo, por realizarse el tratamiento de los datos de forma externa, y ser Unreal Engine únicamente un facilitador para la visualización, se ha elegido el formato tipo "Texto", evitando así incompatibilidades entre separadores decimales, y pudiendo importarse cualquier hoja que contenga información. El nombre que se le ha dado a las variables, para maximizar la compatibilidad es el de los tags del sistema ATAVIA, para integrar los PTL que estén ligados a esta herramienta.

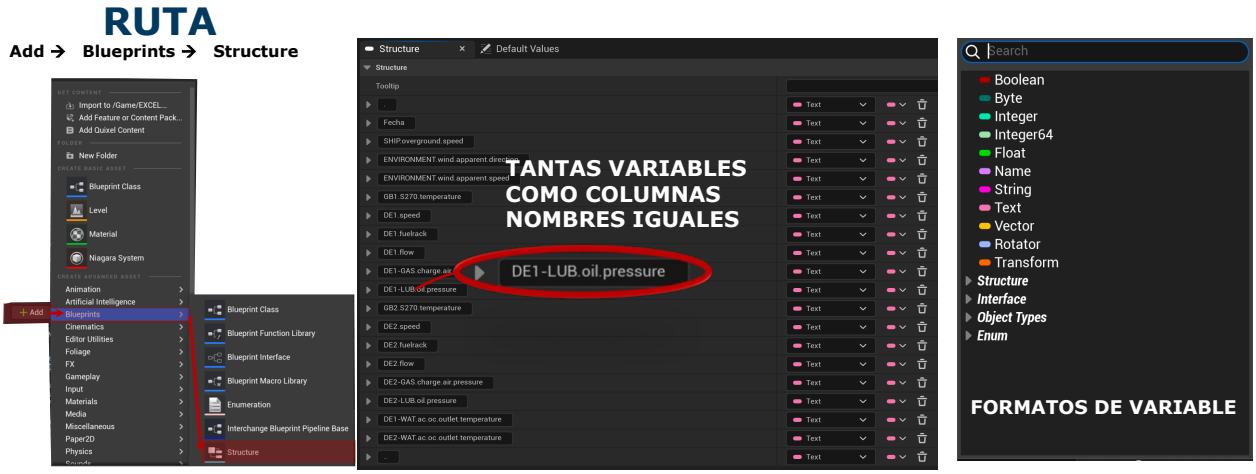


Figura 3-25. Estructura de un *Dataframe*
Elaboración propia

Una vez completado el paso anterior, se importa el documento .csv, al que se le deberá asociar la estructura de *dataframe* definida previamente. El resultado es el que se muestra en la Figura 3-26, una visualización de la tabla con sus valores.

Row	Fecha	SHIP.overground.spee	ENVIRONMENT.wind.apparent.direct	ENVIRONMENT.wind.apparent.spe	GB1.S270.temperatur	DE1.speed	DE1.fuelrack	DE1.flow	DE1-GAS.charge.air.pressu	DE1-LUB.oil.pressur	GB2.S270.te
1	24/10/22 0:00	22,06	5,3	37,36	38	3	1	0	0,04	0,08	44,6
2	24/10/22 0:05	22,06	6,56	38,92	38	3	1	0	0,04	0,08	45
3	24/10/22 0:10	22,08	6,2	39,24	38,6	3	1	0	0,04	0,08	45
4	24/10/22 0:15	22,1	5,8	38,08	39	3	1	0	0,04	0,078	45
5	24/10/22 0:20	22,16	6,24	37,72	39	3	1	0	0,04	0,5	45
6	24/10/22 0:25	22,06	5,66	38,5	39	3	1	0	0,04	0,196	45
7	24/10/22 0:30	21,98	4,96	38,06	39	3	1	0	0,04	0,09	45
8	24/10/22 0:35	21,82	7,84	37,18	39	3	1	0	0,04	0,082	45
9	24/10/22 0:40	21,9	9,26	38,32	39	3	1	0	0,04	0,074	45
10	24/10/22 0:45	21,86	8,5	38,86	39	3	1	0	0,04	0,072	45
11	24/10/22 0:50	22,1	9,06	39,6	39,8	3	1	0	0,04	0,08	45
12	24/10/22 0:55	22,14	9,68	39,48	40	3	1	0	0,04	0,08	45
13	24/10/22 1:00	22,2	9,3	38,96	40	3	1	0	0,04	0,078	45,4
14	24/10/22 1:05	22,24	8,6	37,76	40	3	1	0	0,04	0,08	46

Figura 3-26. *Dataframe* importado a Unreal Engine con estructura definida
Elaboración propia

3.8.5 Desarrollo de los Widgets de visualización del MPD

Al hacer que la cámara se acerque a uno de los componentes con información de la lancha, se debe desplegar un panel con la información que contiene. Para realizar esta acción de mostrar por pantalla la información (que será la contenida en el archivo .csv importado) es posible optar por la metodología de programar un comando *print* o usar un widget, que muestre una interfaz personalizada. Ya que uno de los objetivos de esta rama del proyecto es lograr la inmersión del usuario, se ha desarrollado un widget que simula la interfaz del CAT MPD.

Se debe crear un widget por cada presentación de datos, además de uno que muestre la velocidad del barco al aproximarse a la caña. El widget para los MPD de babor y estribor son iguales, difiriendo en la etiqueta de la banda en la que se ubican y la columna del *dataframe* que tienen asociada.

La Figura 3-27 muestra la interfaz del widget de babor, con una imagen del MPD de elaboración propia, y cuadros de texto correspondientes a cada uno de los datos que se muestran en pantalla, asociados a la función *get_text*, programada desde su propio *blueprint*.

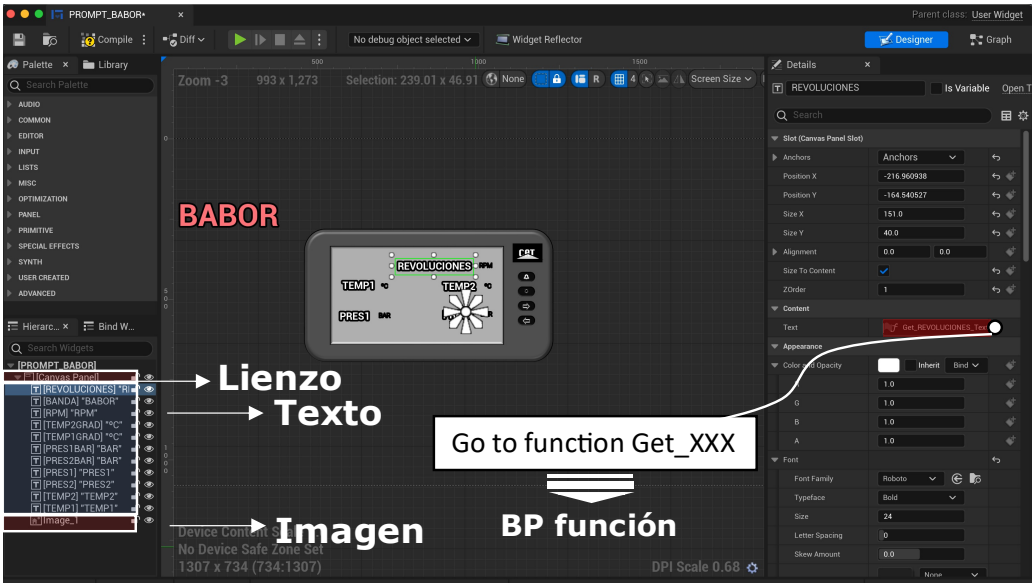


Figura 3-27. Interfaz de Widget en Unreal Engine
Elaboración propia

La función *get_text* ha sido definida para cada caja de texto, ejecutando la misma secuencia en cada caso con la variable de la columna adecuada. Se muestra el correspondiente *blueprint* en la Figura 3-28. Al iniciarse la simulación, el código busca por medio de la función "A" consiste en leer la tabla llamada en este caso "f105" en la fila que sea igual al valor de *row index* (explicado en el apartado 3.8.6.2). Con la función "B" se selecciona la columna de la que se toman los datos. La salida es el almacenamiento como texto (ya que los valores de las columnas se habían definido como tal en la estructura del *dataframe*) de la fila y columna seleccionadas dentro de la variable get_TEMP2 en este caso.

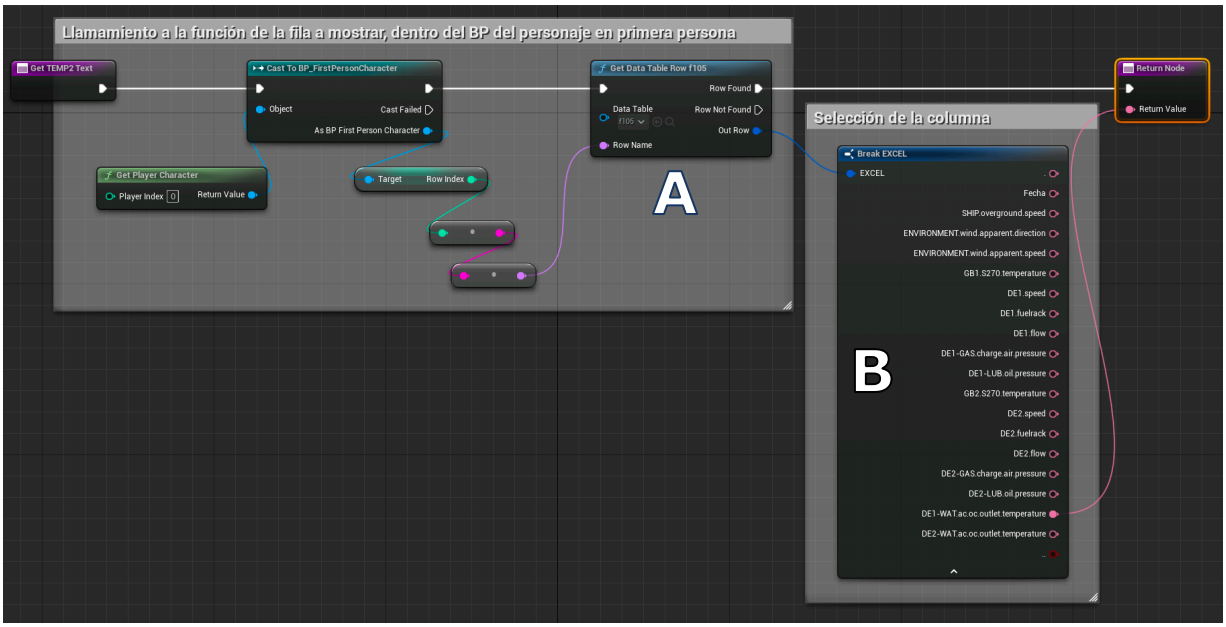


Figura 3-28. Blueprint función "get text"
Elaboración propia

3.8.6 Interacción modelo-Widget

La programación del iniciador del widget se realiza por medio de un *blueprint* de la malla en la que se contiene el modelo 3D objetivo. Una vez creado el nuevo plano, se añaden los elementos de malla estática con la geometría del modelo 3D y una "*Collision box*", redimensionándola y posicionándola donde el usuario pueda atravesarla al acercarse a ella, en un lugar próximo las pantallas de CAT.

Además, el flujo de datos es dinámico desde el inicio de la simulación. Para replicar el avance de los datos, los motores actualizan la información por pantalla cada segundo. Este es el tiempo que debe permanecer el mismo dato almacenado en una variable. El efecto deseado se consigue variando el número de la fila a la que corresponde (*row index*) como se explica más adelante.

3.8.6.1 Colisión

La colisión se configura en el mismo Blueprint del modelo 3D. Al añadir la collision box en cada uno de los MPD, se ejecuta la programación multinodal de la misma para hacer que se muestre el widget del MPD de babor, o de estribor según corresponda.

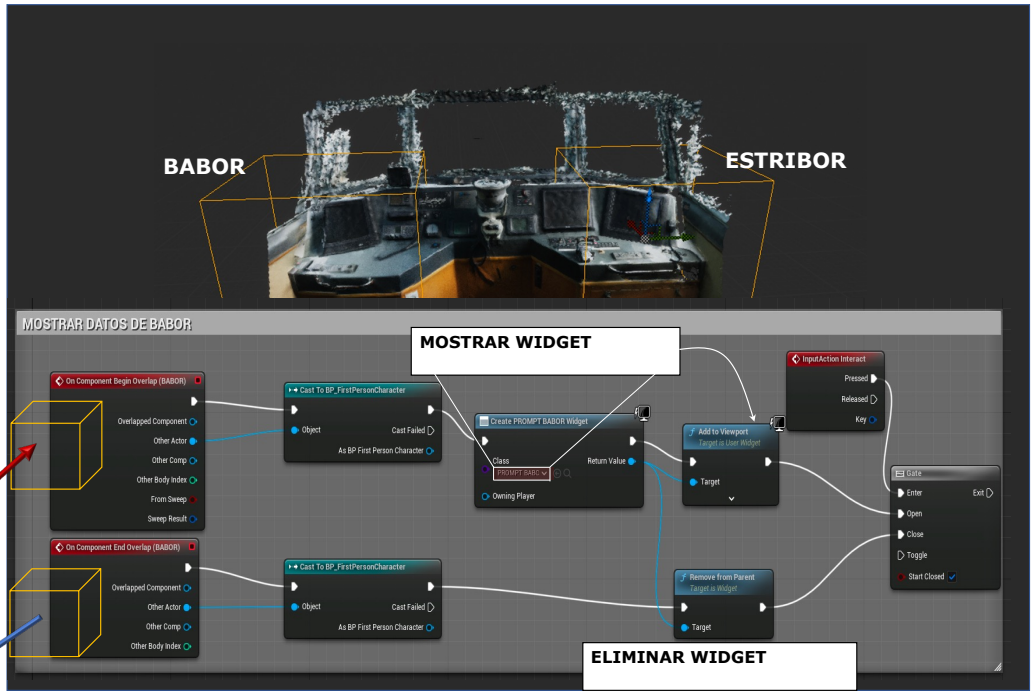


Figura 3-29. Blueprint colisión con MPD de babor
Elaboración propia

En la Figura 3-29 se puede ver un ejemplo de cómo se muestra en pantalla el MPD de babor cuando el usuario entra en los límites de la *collision box* asociada, mostrada como un paralelepípedo con las aristas en color amarillo.

3.8.6.2 Función Row Index

Para la selección de la información a presentar en cada momento, se ha tomado la decisión de actualizar los datos con una frecuencia de un segundo, de la misma forma que lo hace el MPD de la Lancha de Instrucción. Para ello, se ha inicializado un contador de segundos desde la ejecución de la simulación, que luego se asociará a la variable *row index* para indicar qué fila es de la que tiene que mostrar la información. La Figura 3-30 muestra un diagrama de Petri y la programación nodal de esta función. Al iniciarse el evento "*Tick*", equivalente a iniciar la reproducción, se establece un retardo de 1 segundo para ejecutar la acción de la rama inferior, que es sumar al Row index (de tipo *integer*) una unidad. De ahí se obtiene el valor de la variable *Row Index*, a la que se llamaba en Figura 3-28.

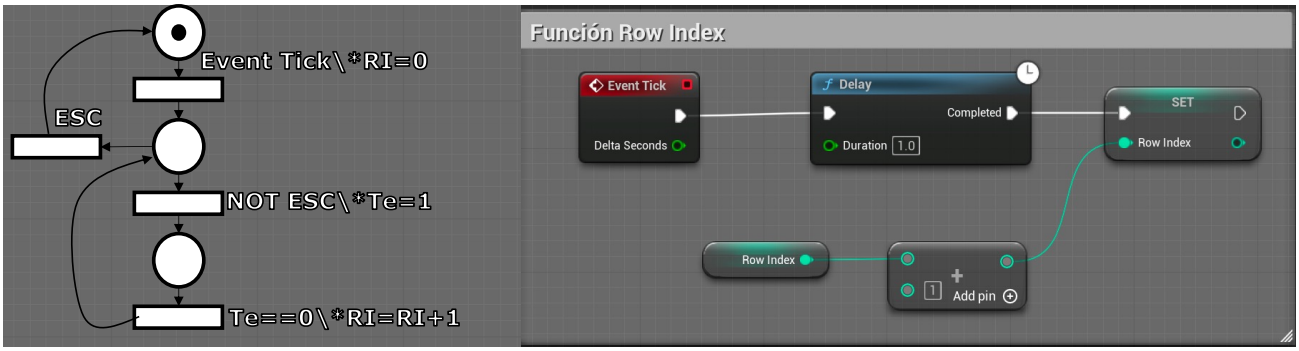


Figura 3-30. Función *Row Index* y diagrama de Petri
Elaboración propia

3.8.7 Desarrollo del programa final

El desarrollo final del proyecto se materializa en una única interfaz, la de la aplicación de Streamlit. Desde ella, es posible realizar los análisis y validaciones de datos pertinentes, así como crear los modelos de buques. La visualización está incorporada por medio del ejecutable descrito en el apartado 3.6.1.3.

Los controles dentro de la visualización inmersiva son los mismos que los definidos en las Generalidades del apartado sobre Unreal Engine.

El resultado es la compilación de una aplicación capaz de tratar y visualizar los datos de operación de los buques. Un entorno capaz de realizar análisis multiplataforma cuya explotación se desarrolla en el capítulo de “Resultados , validación y pruebas realizadas”.

4 RESULTADOS , VALIDACIÓN Y PRUEBAS REALIZADAS

4.1 Generalidades

Se ha dividido esta sección de acuerdo con el alcance de la aplicación desarrollada. Se valoran los aspectos numéricos y la capacidad de análisis de datos, así como la interfaz de la aplicación en sí. Para llevar a cabo la rama de análisis numérico, se ha empleado la propia herramienta para obtener los modelos de la lancha de instrucción A-121 "GM Barrutia", en la que se han realizado las mediciones a bordo de acuerdo con el apartado 3.5.3.

4.2 Resultados de ML

4.2.1 Aplicabilidad de los modelos

En base a los datos tomados a bordo de la lancha de instrucción A-121, se han obtenido los datos que se encuentran anexados al protocolo elaborado (Anexo XII: Documentación operativa para la toma de datos a bordo de las LLII). Con el pretratamiento para la ampliación de valores, explicado en el apartado 3.5.3, se obtiene un conjunto que quintuplica el número de variables inicial, con el que la selección del modelo se realiza de una forma más intuitiva, de forma similar en el caso de contar con material capaz de realizar una medición continua.

Para los modelos obtenidos a bordo de las lanchas de instrucción, debe tenerse en cuenta que, hasta realizar la debida verificación con las demás plataformas de la clase, únicamente serán aplicables a la A-121 "GM Barrutia" por carecer de una línea base en la que establecer sus similitudes y diferencias.

Además, los datos registrados están sometidos a una incertidumbre implícita. En el trabajo de campo, se deben tener en cuenta dos tipos de incertidumbre, una obtenida estadísticamente a partir de las observaciones en las mismas condiciones (incertidumbre de tipo A o S_a), y otras de mayor subjetividad asociadas a la precisión del objeto de medición (incertidumbre tipo B o S_b), en este caso, el Marine Power Display. La suma cuadrática de los dos tipos anteriores permite obtener la incertidumbre combinada o S_c. Las expresiones de cada una de estas variabilidades se muestran en las ecuaciones (21) y (22) para S_a y S_c. respectivamente [48].

$$S_a(\bar{x}) = \frac{\sigma(\bar{x})}{\sqrt{n}} = \frac{\sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}}}{\sqrt{n}} \quad (21)$$

$$S_c = \sqrt{S_a^2 + S_b^2} \quad (22)$$

Para poner a prueba la aplicación desarrollada, se ha decidido elegir tres conjuntos de variables que sean representativos y de especial interés por su aplicación en diferentes campos. Estos son las revoluciones de un mismo eje enfrentadas a la velocidad de la plataforma, el consumo de combustible y la presión a la que está sometida el lubricante. Se realiza en cada una de ellas la correspondiente estimación de la incertidumbre para cada conjunto de datos.

4.2.1.1 Revoluciones frente a velocidad

El estudio de este conjunto de variables permite conocer el avante que se debe ordenar en la plataforma para obtener una velocidad dada en condiciones estándar. Estos resultados son de especial interés en la operación de la plataforma, para navegar a una velocidad dada en situaciones que así lo requieran, como ejercicio de formaciones o de navegación por aguas restringidas con una derrota y protocolo por tiempos definido (recogido en el cuaderno del oficial ayudante de derrota). Además, combinado con el consumo de combustible, explicado en el apartado 4.2.1.2, permite trabajar de una forma más eficiente energéticamente.

En el análisis de estos datos, las incertidumbres calculadas para cada rango de valores se encuentran recogidas por cada grupo de revoluciones del procedimiento "A" en la Tabla 4-1. En ella, para la incertidumbre de tipo B se ha elegido la precisión del equipo ya que el factor humano no condiciona la prueba al haber sido grabada y seleccionado el fotograma del instante con una precisión suficiente.

REVOLUCIONES (RPM)	S_a **(kts)**	S_b **(kts)**	S_c **(kts)**
600	0,03613784	0,1	0,10632941
700	0,04687561		0,11044149
800	0,029646		0,1043019
900	0,03486492		0,10590355
1000	0,04579814		0,1099885
1200	0,04292428		0,10882322
1500	0,14191531		0,17360863

Tabla 4-1. Incertidumbre de la velocidad sobre el fondo.
Elaboración propia

Los resultados obtenidos para cada uno de los rangos de revoluciones deberán tener en cuenta las incertidumbres calculadas, $y \pm S_c$, representadas en la Figura 4-1.

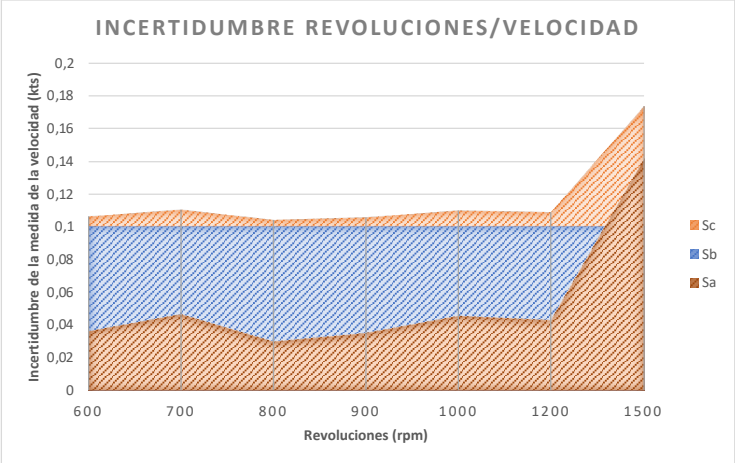


Figura 4-1. Incertidumbres en la medida revoluciones frente a velocidad.
Elaboración propia

Iniciando el tratamiento con la aplicación de *Streamlit*, desde la página "Home", se obtiene la correlación entre ambos resultados, mostrados en la Figura 4-2, con un $r_s = 0{,}97$. Se presenta una fuerte correlación positiva entre ambos valores, a pesar de no ser una proporcionalidad igual a la unidad, lo que refleja el impacto de la incertidumbre debida a las condiciones ambientales.

	SHIP.overground.speed	DE1.speed
SHIP.overground.speed	1.000000	0.971489
DE1.speed	0.971489	1.000000

Figura 4-2. Coeficientes de Spearman revoluciones frente a velocidad.
Elaboración propia

Dentro de la ventana "Data", a la vista de la disposición de los puntos en la gráfica de dispersión, los modelos entrenados generan los polinomios descritos en la Tabla 4-2 y la Figura 4-3.

Grado polinomio	R2 train	R2 test
1	0,93	0,92
2	0,94	0,96
3	0,95	0,96
4	0,96	0,97
5	0,96	0,96
6	0,96	0,96

Tabla 4-2. Resultados generados para revoluciones frente a velocidad.
Elaboración propia

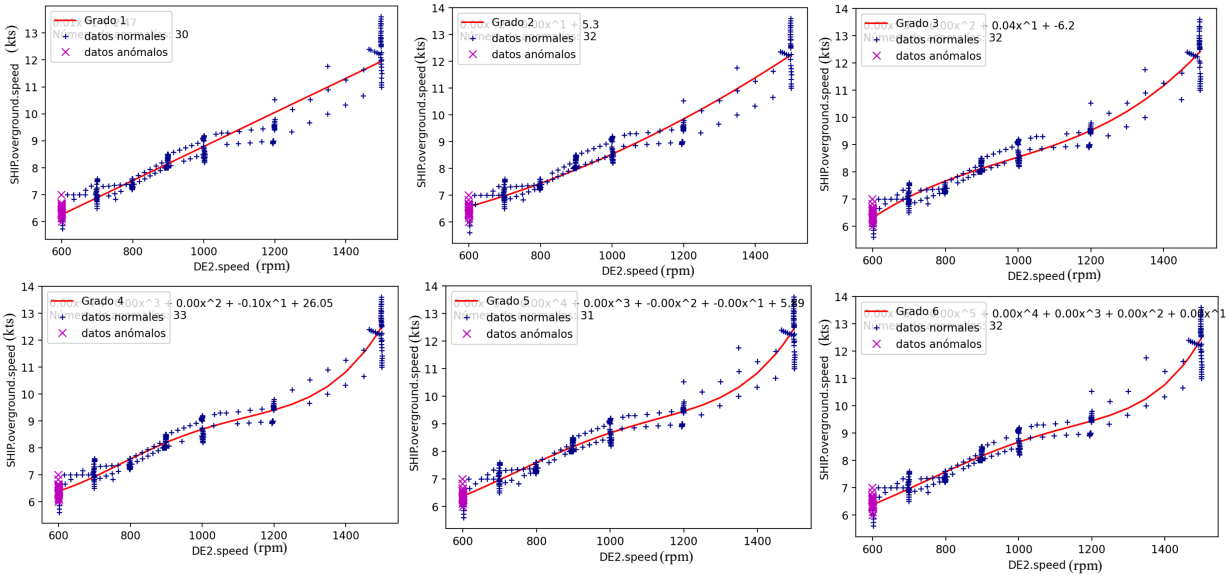


Figura 4-3. Gráficas de los polinomios generados para revoluciones frente a velocidad.
Elaboración propia, velocidad en nudos, revoluciones en vueltas por minuto

Dado su buen ajuste, validado por el coeficiente de determinación R^2, así como la trayectoria seguida por la curva, se ha elegido el polinomio de grado 4 como el más representativo para estas dos variables. En la Figura 4-4 se muestra el resultado de este modelo, así como el intervalo de confianza del 95 %. Cabe destacar que la tendencia de la incertidumbre combinada S_c es creciente a medida que aumentan las revoluciones, por lo que la mayor variabilidad de datos se dará en órdenes de avante más rápidas.

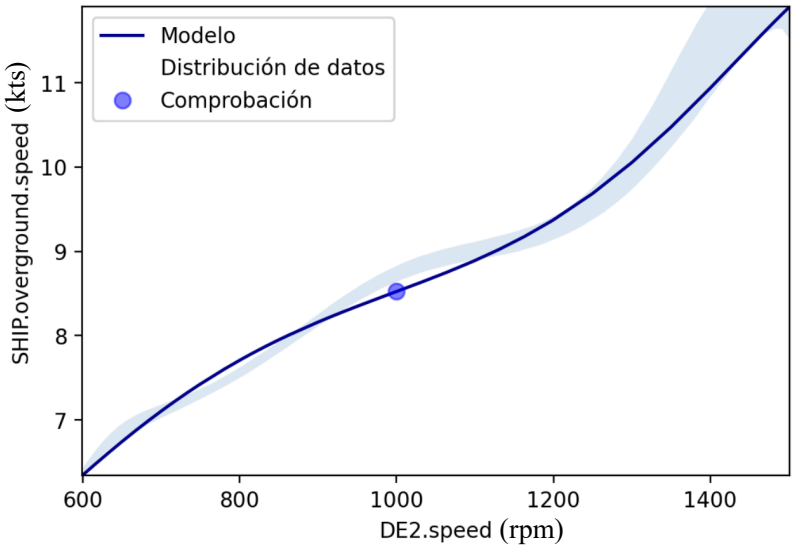


Figura 4-4. Modelo revoluciones frente a velocidad sobre el fondo.
Elaboración propia, velocidad en nudos, revoluciones en vueltas por minuto

Sobre este modelo pueden representarse las predicciones para ciertas revoluciones y los parámetros del comprobador de anomalías. El círculo representado como "comprobación" corresponde a la

predicción para 1000 revoluciones por minuto, con una salida de 8,54±0,109 nudos[10] (teniendo en cuenta la incertidumbre combinada calculada anteriormente).

Sobre la visualización inmersiva para este modelo, se presenta un diagrama con los resultados obtenidos en la Figura 4-5.

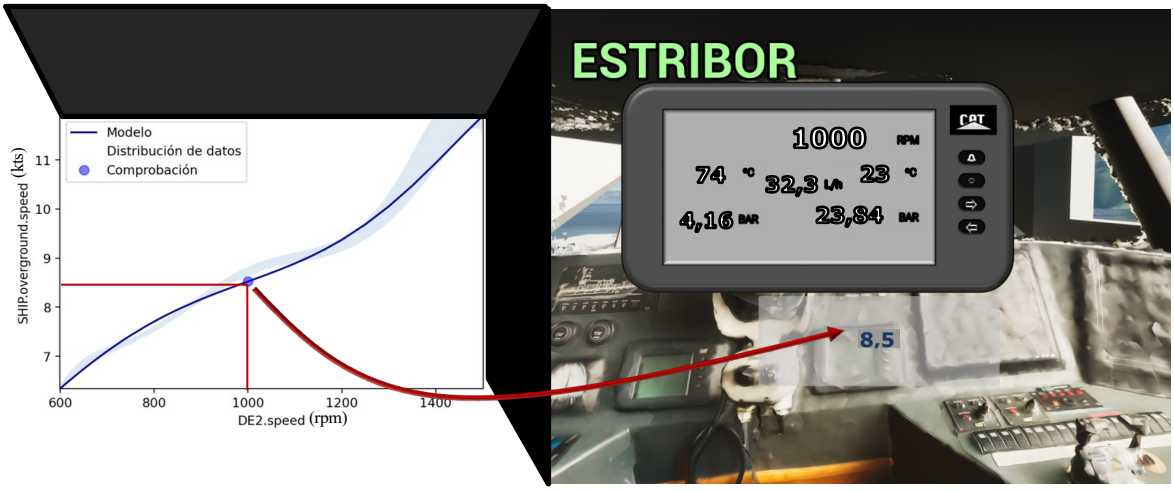


Figura 4-5. Visualización inmersiva del modelo para revoluciones y velocidad sobre el fondo.
Elaboración propia

4.2.1.2 Revoluciones frente a consumo de combustible

El estudio de estos dos valores muestra la demanda de combustible en la operación de estas unidades, siendo posible estimar su consumo en un periodo determinado de tiempo, así como diversas aplicaciones en pos del aprovechamiento de los combustibles fósiles y la eficiencia energética.

En el análisis de estos datos, las incertidumbres calculadas para cada rango de valores se encuentran recogidas por cada grupo de revoluciones del procedimiento "A" en la Tabla 4-3. Para la incertidumbre de tipo B se ha elegido el valor de 3 L/h de combustible por ser el mínimo apreciable con cierta confianza en la escala representada en el MPD, en intervalos de 14 L/h y marcas menos cada 7 L/h (el máximo error que se cometería en el entorno del análisis de fotogramas será de $\pm\frac{1}{2}$ del intervalo menor).

REVOLUCIONES (RPM)	S_a (L/h)	S_b (L/h)	S_c (L/h)
600	0,13384595	3	3,00298431
700	0,23191753		3,00895094
800	0,26492511		3,01167484
900	0,25978795		3,01122729
1000	0,3391194		3,01910615
1200	0,55341639		3,05061792
1500	0,38929839		3,02515342

Tabla 4-3. Incertidumbre del consumo de combustible.

[10] Este valor se emplea más adelante en el procedimiento de prueba con personal externo descrito en el apartado 4.3.1

Elaboración propia

Los resultados obtenidos para cada intervalo de revoluciones tendrán en cuenta las incertidumbres calculadas, $y \pm S_c$, representadas en la Figura 4-6.

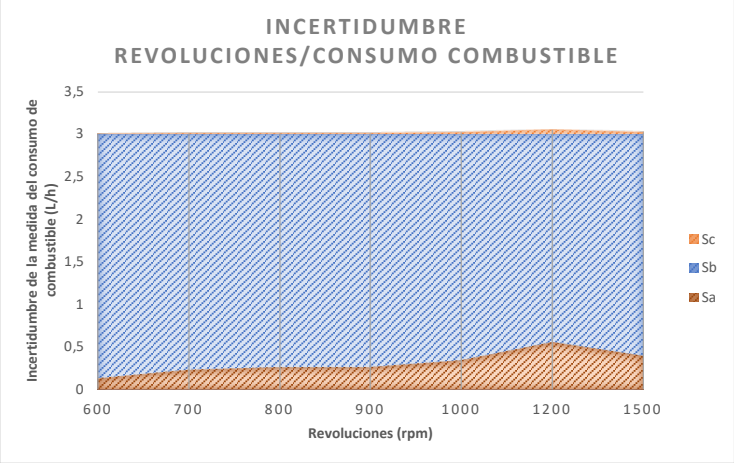


Figura 4-6. Incertidumbres en la medida revoluciones frente a consumo de combustible.
Elaboración propia

De la misma forma que en el caso anterior, iniciando el tratamiento desde la aplicación de Streamlit, en la página "Home", se obtiene la correlación entre ambos resultados, mostrados en la Figura 4-7, con un $r_s = 0{,}96$. En este caso, también existe una marcada correlación positiva entre ambos valores, a pesar de no ser un valor de 1, observando gradientes mayores de consumo a medida que las revoluciones aumentan, no siendo lineal.

	SHIP.overground.speed	DE1.speed	DE1.fuelrack
SHIP.overground.speed	1.000000	0.971489	0.960103
DE1.speed	0.971489	1.000000	0.976630
DE1.fuelrack	0.960103	0.976630	1.000000

Figura 4-7. Coeficientes de Spearman revoluciones frente a consumo de combustible.
Elaboración propia

Dentro de la ventana "Data", a la vista de la disposición de los puntos en la gráfica de dispersión, los modelos entrenados generan los polinomios descritos en la Tabla 4-4 y la Figura 4-8.

Grado polinomio	R2 train	R2 test
1	0,97	0,95
2	0,97	0,98
3	0,97	0,98
4	0,97	0,97

Grado polinomio	R2 train	R2 test
5	0,97	0,98
6	0,96	0,97

Tabla 4-4. Resultados generados para revoluciones frente a consumo de combustible.
Elaboración propia

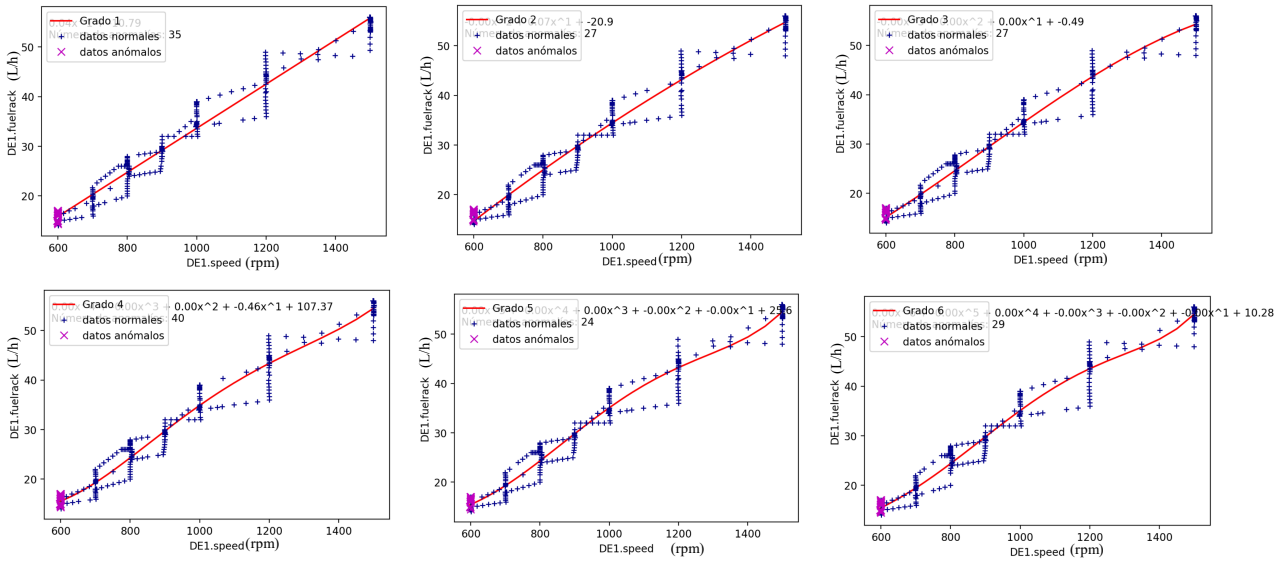


Figura 4-8. Representación de de modelos revoluciones / consumo de combustible.
Elaboración propia, consumo en litros por hora, revoluciones en vueltas por minuto

Dado su ajuste, validado por el coeficiente de determinación R^2, así como la trayectoria seguida por la curva, se ha elegido el polinomio de grado 5 como el más representativo para estas dos variables. A pesar de ajustarse también con fidelidad los modelos de menor grado, es en este primero en el que se puede apreciar una diferencia más marcada de los gradientes, presentando una menor variación de datos intermedio (entre las 1000 y 1200 revoluciones) y, aproximándose a los extremos mayores últimas, un aumento de la pendiente gradual.

En la Figura 4-9 se muestra el resultado de este modelo, así como el intervalo de confianza del 95 por ciento. Cabe destacar que la tendencia de la incertidumbre combinada S_c es creciente a medida que aumentan las revoluciones, por lo que la mayor variabilidad de datos de dará en órdenes de avante más rápidas.

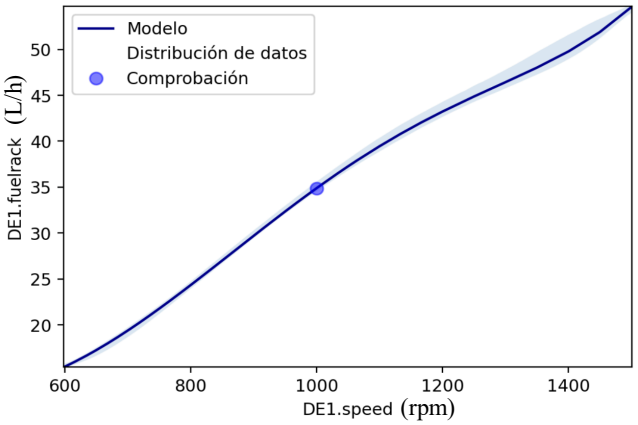


Figura 4-9. Modelo revoluciones frente a consumo de combustible.
Elaboración propia, consumo en litro por hora, revoluciones en vueltas por minuto

Sobre este modelo pueden representarse las predicciones para ciertas revoluciones y los parámetros del comprobador de anomalías. El círculo representado como "comprobación" corresponde a la predicción para 1000 revoluciones por minuto, con una salida de 35,03±3,019 litros por hora (teniendo en cuenta la incertidumbre combinada calculada anteriormente). La incertidumbre representa cerca de un 8,5 % del valor estimado, elevada debido al error de medida que se estima puede haberse cometido. Una posibilidad para la mejora del tratamiento es la obtención de datos numéricos directamente del ordenador con el CAT COMM III, explicado con más detalle en el apartado 5.2.1.

Sobre la visualización inmersiva para este modelo, se presenta un diagrama con los resultados obtenidos en la Figura 4-10.

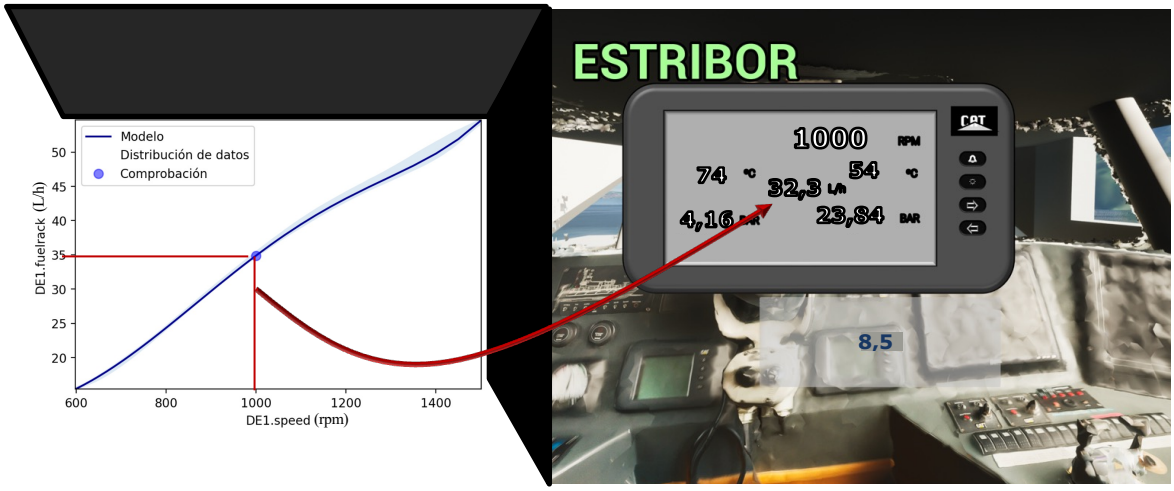


Figura 4-10. Visualización inmersiva del modelo para revoluciones y consumo de combustible
Elaboración propia

El estudio de estas variables permite analizar el consumo para un periodo de tiempo, conociendo las necesidades de maniobra de la plataforma. Se muestran a continuación los casos de tres ejercicios comunes a bordo de las lanchas de instrucción:

- Ejercicio de navegación por aguas restringidas: la plataforma debe maniobrar sobre una derrota establecida. Para facilitar el posicionamiento sobre la misma, es aconsejable realizarla a una velocidad predeterminada, sabiendo los tiempos a navegar en cada tramo. En los ejercicios llevados a cabo en la ría de Pontevedra, la longitud de las derrotas es de media 7 millas náuticas, de acuerdo con la documentación operativa de la asignatura de adiestramiento a flote en el campus virtual corporativo de defensa. La Tabla 4-5 muestra un análisis de los consumos que se darían en un ejercicio de este tipo según su velocidad. Las revoluciones para cada velocidad se han obtenido del modelo generado en el apartado anterior, siendo de uno de los dos motores, por lo que, para obtener el consumo completo, es necesario llevar a cabo de nuevo este proceso de análisis con los equipos de babor y sumar ambos resultados. El consumo total se obtiene de la multiplicación del valor de consumo por hora del modelo generado por la duración de la navegación.

Velocidad (kts)	Ejercicio	Revoluciones (rpm)	Duración para 7MN	Consumo total de combustible
6	Iniciación y condiciones de baja visibilidad	600	1 h 10 m	17,97 L

Velocidad (kts)	Ejercicio	Revoluciones (rpm)	Duración para 7MN	Consumo total de combustible
7	Condiciones normales/nocturnas	715	1 h	20,04 L
8		871	52 m	24,45 L
9	Condiciones normales diurnas	1085	46 m	39,32 L

Tabla 4-5. Cálculo de consumos en ejercicios de navegación por aguas restringidas.
Elaboración propia, motor principal de estribor

- Ejercicio de aproximaciones de aprovisionamiento: una de las unidades simula una plataforma proveedora, manteniéndose a rumbo y velocidad constante (7 nudos en las lanchas de instrucción habitualmente) mientras el buque receptor debe aproximarse por su costado desde una posición atrasada y mantenerse través con través durante el tiempo establecido en el ejercicio. Una vez finalizado, el buque receptor aumenta velocidad y cae levemente a la banda contraria a la que se encuentra el buque proveedor. La duración total de este procedimiento es de cerca de 8 minutos para los movimientos entre posiciones sumado al tiempo que se permanezca estático en el través del otro buque. La Tabla 4-6 muestra el cálculo del consumo medio de combustible para una aproximación, debiendo este resultado multiplicarse por la duración de esta para obtener el consumo total.

Velocidad (kts)	Acción / % de tiempo	Revoluciones (rpm)	Consumo de combustible por hora
7	Estacionario adoptando posición inicial a popa del proveedor / 20 %	715	20,04 L/h x 0,2
11	Aproximación a la banda del buque proveedor / 15 %	1415	51,04 L/h x 0,15
7	Mantenimiento de la posición través con través / 50 %	715	20,04 L/h x 0,5
13	Alejamiento del buque proveedor aumentando velocidad / 15 %	1525	55,96 L/h x 0,5
		Consumo total	49,6 L/h x $t_{aproximación}$

Tabla 4-6. Cálculo de consumos en ejercicios de aproximaciones de aprovisionamiento.
Elaboración propia

- Ejercicio de evoluciones simples: formaciones ejecutadas por la agrupación de lanchas de acuerdo con las instrucciones de maniobra de la doctrina OTAN. Normalmente, se opera con tres velocidades, una de ejecución de la formación (N Speed), otra de evolución para ocupar los puestos asignados (S Speed) y otra de emergencia para los casos especiales que puedan requerirla (X Speed). Habitualmente, los valores de estas velocidades son 7, 11 y 13 nudos en los ejercicios de la Escuela Naval para la N, S y X Speed respectivamente. La tabla Tabla 4-7 muestra el cálculo realizado de consumo medio por hora para un ejercicio de este tipo. Se han establecido unos porcentajes de tiempo a cada una de las velocidades mencionadas anteriormente para aplicar una media ponderada al resultado.

Velocidad	Porcentaje de tiempo en operación	Revoluciones (rpm)	Consumo de combustible por hora
7 (N)	60 %	715	20,04 L/h x 0,6
11 (S)	35 %	1415	51,04 L/h x 0,35
13 (X)	5 %	1525	55,96 L/h x 0,05
		Consumo total	32,686 L/h

Tabla 4-7. Cálculo de consumos en ejercicios de evoluciones.
Elaboración propia

4.2.1.3 Revoluciones a presión del aceite de lubricación

En el mantenimiento de las unidades, un factor de especial relevancia es el estado del lubricante que actúa en los motores. Un deterioro de este implica una mayor probabilidad de fallo en el sistema completo, pudiendo desencadenar la rotura o inoperatividad del equipo. El conocimiento del estado del lubricante, presión y temperatura es una de las claves del mantenimiento preventivo y, sobre todo, del predictivo.

En el análisis de estos datos, las incertidumbres calculadas para cada rango de valores se encuentran recogidas por cada grupo de revoluciones del procedimiento "A" en la Tabla 4-8. En ella, para la incertidumbre de tipo B se ha elegido la precisión del equipo ya que el factor humano no condiciona la prueba al haber sido grabada y seleccionado el fotograma del instante con una precisión suficiente.

REVOLUCIONES (RPM)	S_a (bar)	S_b (bar)	S_c (bar)
600	1,97769054	1	2,21613625
700	2,26537521		2,47627237
800	2,18879471		2,40641274
900	5,34482183		5,43756567
1000	4,57790788		4,68585537
1200	3,2109501		3,36306416
1500	2,04790475		2,27901599

Tabla 4-8. Incertidumbre de la presión del aceite de lubricación.
Elaboración propia

Los resultados obtenidos para cada intervalo de revoluciones tendrán en cuenta las incertidumbres calculadas, $y \pm S_c$, representadas en la Figura 4-11.

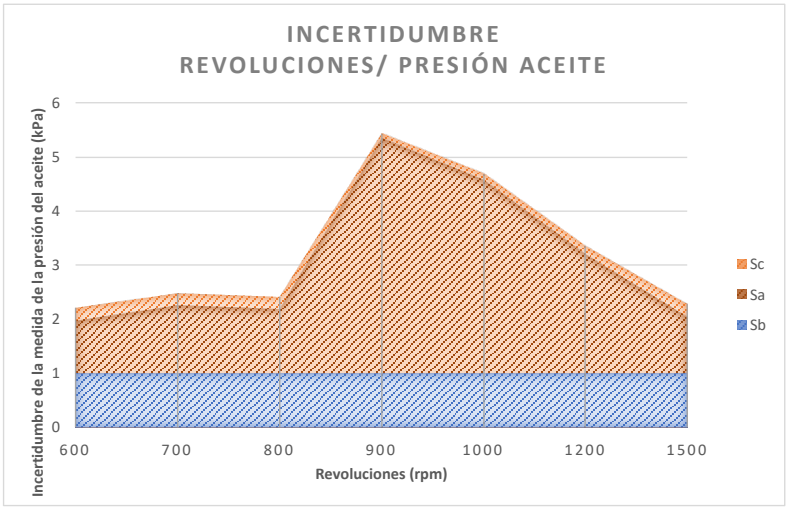


Figura 4-11. Incertidumbres en la medida revoluciones frente a presión del aceite de lubricación.
Elaboración propia

De la misma forma que en el caso anterior, iniciando el tratamiento desde la aplicación de Streamlit, en la página "Home", se obtiene la correlación entre ambos resultados, mostrados en la Figura 4-12,con un $r_s = 0{,}964$. En este caso, también existe una marcada correlación positiva entre ambos valores.

	SHIP.overground.speed	DE1.speed	DE1.fuelrack	DE1.flow	DE1-GAS.charge.air.pressure	DE1-LUB.oil.pressure
SHIP.overground.speed	1.000000	0.971489	0.960103	0.931458	0.951679	0.949723
DE1.speed	0.971489	1.000000	0.976630	0.944830	0.959948	0.964334
DE1.fuelrack	0.960103	0.976630	1.000000	0.973898	0.943143	0.968227
DE1.flow	0.931458	0.944830	0.973898	1.000000	0.899753	0.946515
DE1-GAS.charge.air.pressure	0.951679	0.959948	0.943143	0.899753	1.000000	0.949486
DE1-LUB.oil.pressure	0.949723	0.964334	0.968227	0.946515	0.949486	1.000000

Figura 4-12. Coeficientes de Spearman revoluciones a presión del aceite de lubricación.
Elaboración propia

Dentro de la ventana "Data", a la vista de la disposición de los puntos en la gráfica de dispersión, los modelos entrenados generan los polinomios descritos en la Tabla 4-9 y la Figura 4-13.

Grado polinomio	R2 train	R2 test
1	0,87	0,93
2	0,94	0,95
3	0,94	0,96
4	0,95	0,90
5	0,95	0,94
6	0,94	0,96

Tabla 4-9. Resultados generados para revoluciones frente a presión del aceite de lubricación.
Elaboración propia, presión en kilopascales, revoluciones en vueltas por minuto

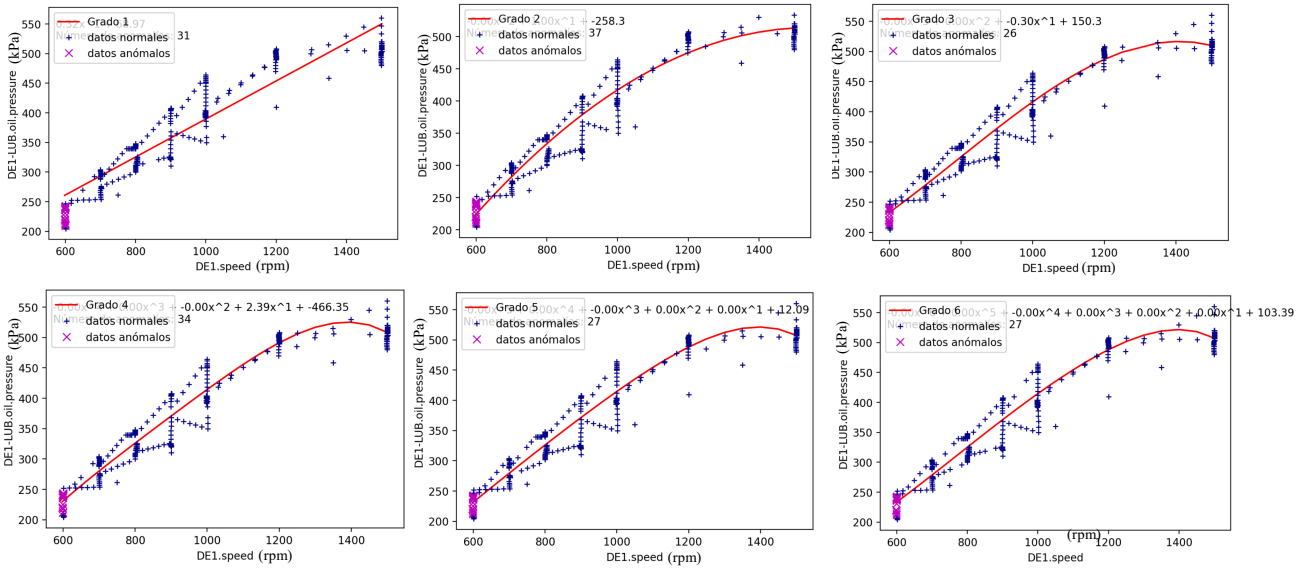


Figura 4-13. Representación de modelos revoluciones / presión del aceite de lubricación.
Elaboración propia, presión en kilopascales, revoluciones en vueltas por minuto

Dado su ajuste, validado por el coeficiente de determinación R^2, así como la trayectoria seguida por la curva, se ha elegido el polinomio de grado 5 como el más representativo para estas dos variables. A pesar de ajustarse también con fidelidad los modelos de menor grado, es en este primero en el que se puede apreciar una diferencia más marcada de los gradientes, presentando una menor variación de datos intermedio (entre las 1000 y 1200 revoluciones) y, aproximándose a los extremos mayores últimas, un aumento de la pendiente gradual.

En la Figura 4-14 se muestra el resultado de este modelo, así como el intervalo de confianza del 95 %. Cabe destacar que la tendencia de la incertidumbre combinada S_c es creciente a medida que aumentan las revoluciones, por lo que la mayor variabilidad de datos se dará en órdenes de avante más rápidas.

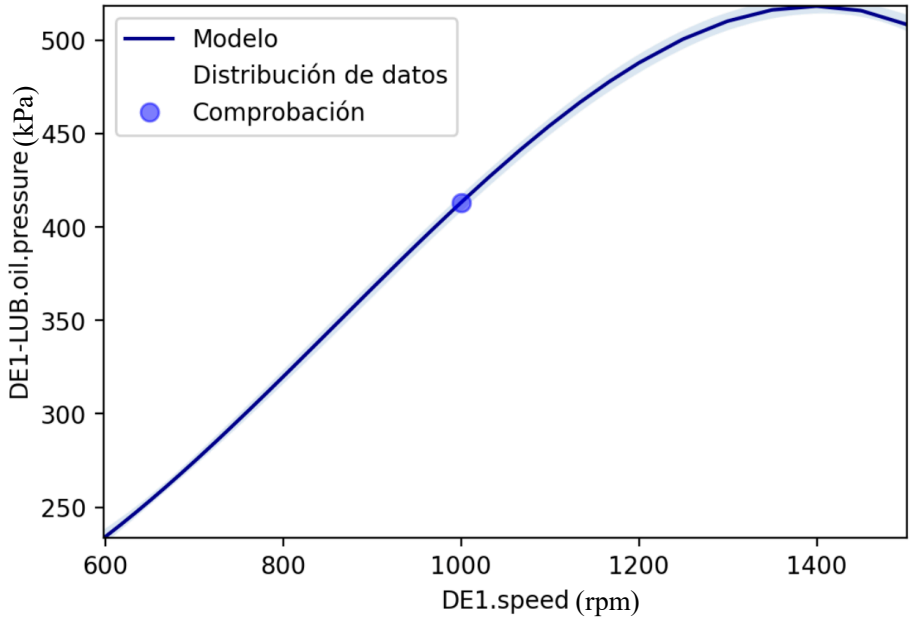


Figura 4-14. Modelo revoluciones frente a presión del aceite de lubricación.
Elaboración propia, presión en kilopascales, revoluciones en vueltas por minuto

De la misma manera que en los casos anteriores, se pueden representar las entradas de predicción y comprobador de anomalías. El círculo representado como "comprobación" corresponde a la predicción para 1000 revoluciones por minuto, con una salida de 413±4,68 kilopascales (teniendo en cuenta la incertidumbre combinada calculada anteriormente).

Sobre la visualización inmersiva para este modelo, se presenta un diagrama con los resultados obtenidos en la Figura 4-15.

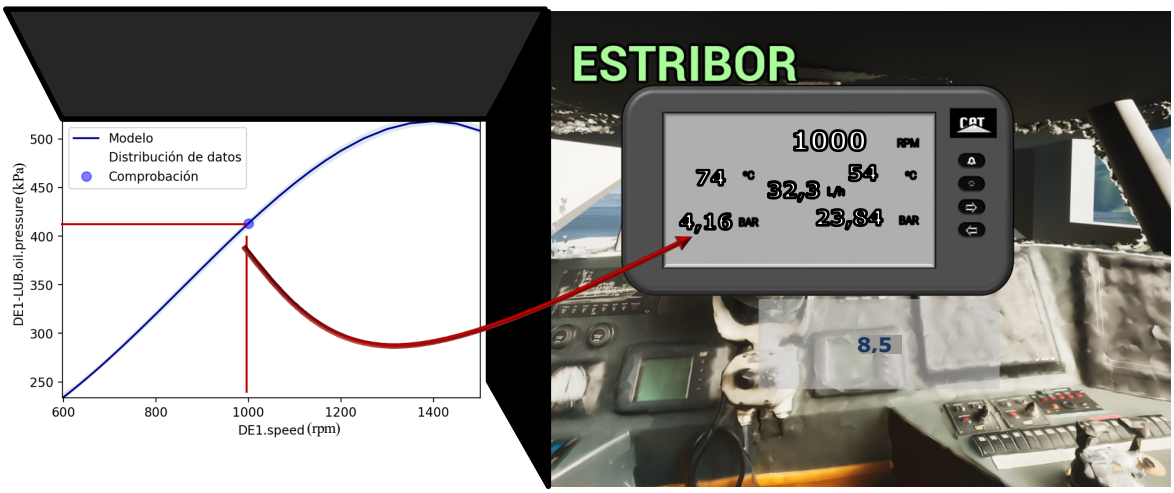


Figura 4-15. Visualización inmersiva del modelo revoluciones / presión del aceite.
Elaboración propia

4.3 Explotación de la aplicación

El software desarrollado tiene la característica de que su autor podría ser uno de sus usuarios en el futuro, en destinos dedicados al sostenimiento de las unidades. Es por ello, que se ha creado desde el conocimiento de las necesidades que debe satisfacer como herramienta de apoyo a la fuerza.

4.3.1 Grado de usabilidad

La palabra usabilidad es un neologismo no reconocido todavía por la Real Academia Española pero que, por su carácter tecnológico es empleada en el ámbito del diseño de interfaces. La estricta definición está recogida en la norma ISO 25010, siendo una capacidad del desarrollo software para garantizar su entendimiento y atractividad en el uso. La normativa introduce términos en los que engloba la subcaracterísticas de esta propiedad [49]:

- Reconocibilidad de la adecuación: la facilidad con la que el personal es capaz de entender las limitaciones y el grado de cobertura de sus necesidades por la aplicación.
- Aprendizabilidad: capacidad de aprendizaje del público del software.
- Operabilidad: característica de la facilidad de interacción público-aplicación.
- Protección contra errores del público: capacidad de la aplicación para prevenir, sea detectando o tratando los errores que puedan ser introducidos por el público de la aplicación.
- Estética de la interfaz de usuario: agrado que siente el público por el diseño del software.
- Accesibilidad: capacidad de adaptación al personal empleador, dadas sus características particulares.

Para comprobar la cobertura de estas necesidades, se le ha presentado la aplicación a un aspirante de segundo de cuerpo general (alumno de segundo curso) de la Escuela Naval Militar (Figura 4-16), cuyas funciones a bordo de las lanchas de instrucción son equiparables a las que desempeña un suboficial a bordo. En el caso de su papel en puestos como supervisor de máquinas, debe velar por el buen

funcionamiento de equipos y sistemas embarcados, tarea que podría simplificarse mediante una de las posibles líneas futuras explicadas más adelante, sobre la toma de decisiones en materia de sostenimiento.

Para la validación de los factores de la usabilidad reglados por la normativa ISO, se ha analizado el caso del alumno mencionado anteriormente de acuerdo con el siguiente procedimiento:

1. El alumno ha sido guiado por el autor en el manejo de los datos y su visualización en ambos entornos, tanto el numérico como el inmersivo.
2. Se ha proporcionado acceso al manual de usuario descrito en el apartado 4.3.2 para consulta durante la prueba.
3. Una vez realizada la toma de contacto, se procede a la realización de una prueba consistente en resolver el siguiente problema:
   El MPD de la lancha A-121 "GM Barrutia" muestra que, para un avante de 1000 rpm, con ambos motores, la velocidad sobre el fondo es de 7 nudos. Genere el modelo correspondiente a estos parámetros con los datos recogidos en la navegación del 14 de marzo de 2023 y compruebe si estos datos se ajustan al modelo y, en caso contrario, indique el valor esperado proporcionado por el modelo construido. Visualice también los datos en el entorno inmersivo". Como resultado, se toma como válido el modelo generado en el apartado 4.2.1.1, con un resultado de 8,54 nudos esperado en un avante 1000.

El alumno debe cubrir el proceso de:

- Selección del archivo y hoja que contiene los datos.
- Selección de parámetros a analizar (revoluciones frente a velocidad sobre el fondo) y periodo temporal correspondiente a la salida a la mar del 14 de marzo.
- Introducir velocidad de ralentí del motor y selección del tipo de plataforma como "lancha de instrucción".
- Selección de un grado del modelo que se tenga un buen ajuste en la zona a observar, actuando si fuese necesario, sobre los filtros por cuartiles.
- Introducir los datos del enunciado en el comprobador de anomalías y en el cajetín de predicción.
- Una vez finalizado el análisis numérico, seleccionar la página visualización y ejecutar el programa, desplazándose hasta la collision box de los MPD y estando lo suficientemente cerca de la caña para mostrar también los datos de velocidad.



Figura 4-16. Prueba de la aplicación por personal ajeno al trabajo[11].
Elaboración propia

[11] Se cuenta con la aprobación del alumno para la inclusión de la imagen, aunque en virtud del apartado 3 del capítulo I de la LO 1/1982 no se incluirá información personal del mismo por decisión del autor.

La prueba realizada se completa satisfactoriamente, alcanzando los objetivos de la aplicación. El alumno es capaz de resolver el problema y de ver los datos del día seleccionado en la interfaz inmersiva de Unreal Engine 5, por lo que se cumplen los objetivos de facilidad de aprendizaje y de operabilidad.

Los resultados hacen reflexionar al alumno sobre el causante de presentar una velocidad menor, lo que desencadenaría, en un escenario real, el comienzo de una investigación para la aplicación de medidas correctivas, deducciones que verifican el suficiente conocimiento de la adecuación del software.

El alumno introduce las revoluciones de ralentí del motor para poder visualizar los datos, así como entiende el funcionamiento de las demás herramientas Poka-Yoke, evitando así cometer errores en la selección de datos.

La interfaz se considera amigable por su diseño claro y conciso, y puede ser visualizada con facilidad haciendo zoom en caso de ser necesario en cualquiera de los gráficos o datos numéricos. Es accesible en este sentido, a pesar de no tener en consideración otro tipo de limitaciones o discapacidades a la hora de operar con ella. El personal que trabajaría con esta herramienta, por lo general, estará sujeto a los requisitos psicofísicos de las fuerzas armadas, características que no requieren ningún tratamiento especial del diseño de la aplicación salvo modificaciones al tamaño de la fuente o de los elementos por limitaciones de visión leves, cambios que puede realizarse desde el menú de la propia aplicación que ofrece *Streamlit* para personalización secundaria del diseño.

4.3.2 Guía de empleo

Siendo la aplicación una herramienta multiplataforma, puede ser compartida a las diferentes unidades para su inclusión en las tareas de vigilancia y análisis. Para facilitar la adecuación al software, se ha decidido elaborar un manual de usuario, recogido en forma de vídeo y que puede ser accesible desde el enlace: https://youtu.be/_BbzPxjLNFY

También se encuentra el acceso a este archivo por medio del código QR mostrado en y en el resumen gráfico de este trabajo:

Figura 4-17. Código QR con enlace a vídeo del manual de empleo.
Elaboración propia

.

5 CONCLUSIONES Y LÍNEAS FUTURAS

5.1 Conclusiones del trabajo

Al inicio del trabajo, se plantea la necesidad de una plataforma de vigilancia y análisis de datos con la que poder llevar a cabo labores de apoyo al CdV de las unidades, así como otras derivadas. La aplicación desarrollada pretende servir como herramienta para el mantenimiento de las unidades de la Escuela Naval, y servir como instrumento de apoyo en la formación de oficiales, introduciendo el ámbito del sostenimiento en su enseñanza de formación.

En esta sección se describe el alcance final del proyecto contrastado con los objetivos fijados al inicio de este. También, se dividen las líneas futuras aplicables en tres ámbitos: el académico, el industrial y el del Ministerio de Defensa; por su naturaleza investigadora, de explotación y concordancia con la estrategia de tecnología e innovación de la DGAM, respectivamente.

5.1.1 Grado de satisfacción de los objetivos iniciales

Los objetivos establecidos en el apartado 1.2 se dividen en principales y secundarios, estando los primeros orientados al asentamiento de las bases de la aplicación y, los segundos, a mejoras sobre éstos.

De acuerdo con los objetivos principales, sí que se satisfacen los requisitos de creación de una herramienta capaz de modelar el comportamiento de los buques en base a los datos observados, gracias a las tecnologías de Machine Learning (uno de los objetivos secundarios). La interfaz presentada tiene capacidad de interacción del personal investigador para el desarrollo de modelos personalizados, con los filtros que sean considerados de interés. Además, esta interfaz viene complementada con la inmersión en el entorno de la lancha de instrucción A-121 “GM Barrutia”, leyendo los datos generados por los modelos anteriores, satisfaciendo el objetivo secundario de la interacción con un modelo tridimensional.

5.2 Líneas académicas

Las líneas futuras en el ámbito académico han sido ideadas con el alumnado de Escuela Naval Militar como principal personal trabajador. Por una parte, se presentan las ideas relativas a su mejora, actualización y desarrollo, posibles líneas de investigación en trabajos. A continuación, el hecho de contar con esta herramienta precisa de un mantenimiento de sus capacidades, el apoyo a su CdV, se detallan los requisitos a completar para lograr el máximo aprovechamiento de la aplicación actual.

5.2.1 Mejoras

Este trabajo es un ejemplo de las bases que un programa de monitorización de variables de buques podría presentar, pudiendo ser sujeto de diversas mejoras que optimicen su potencial. Como herramienta

del apoyo a la fuerza, debe regirse por la "Visión del AJAL del Arsenal inteligente" de enero de 2023, de acuerdo con la prerrogativa de buscar el avance de una organización colaborativa a una inteligente, la búsqueda de la mejora continua. Para la actualización de este proyecto, se plantean las líneas de desarrollo en materia de tratamiento de datos y de visualización:

- Líneas futuras en el tratamiento de los datos:
  1. Estudio de la implementación de algoritmos no supervisados de ML de clasificación para la detección de anomalías en el funcionamiento. Como ejemplo, el empleo del algoritmo de Support Vector Machine de clase única, para la detección de estos comportamientos alejados de los modelos de operación
  2. Análisis de opciones de transferencia de datos en directo desde la plataforma a la aplicación, como alternativa a la sensorización, a través de un servidor y la adquisición del conector CAT COMM III.

- Las líneas futuras en el campo de la visualización inmersiva son:
  1. Dotar al programa de visualización de un entorno de simulación, con movilidad de la plataforma en el escenario creado de la ría de Pontevedra, con monitorización de las variables de posición GPS en directo, por medio de datos obtenidos con la línea número 2 de tratamiento de datos, o con información AIS en directo.
  2. Adecuación del escenario a la visualización con realidad virtual, para su uso a distancia, y realidad aumentada, para su uso a bordo para labores de mantenimiento embarcado.
  3. A partir de la línea futura de visualización número 2, desarrollo de un simulador de las lanchas, que permita el adiestramiento del alumnado tanto en el desempeño de la maniobra en la plataforma, como en las acciones a tomar en caso de fallos a bordo. Una herramienta de apoyo formación en la toma de decisiones.

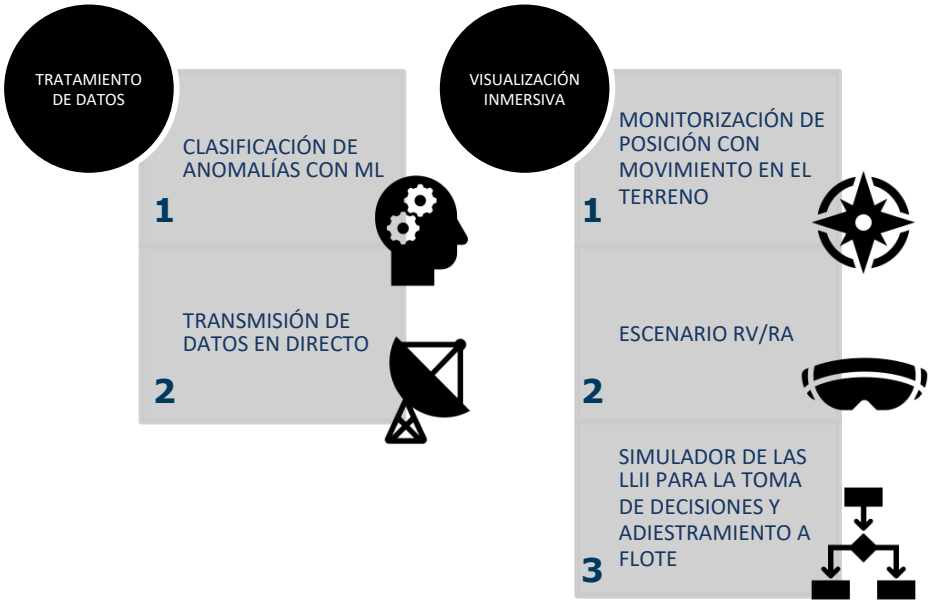


Figura 5-1.Resumen líneas futuras de desarrollo académicas
Elaboración propia

5.2.2 Ciclo de Vida del software

Como cualquier desarrollo, el software está sujeto a unas condiciones de deterioro o de pérdida de valor en caso de un mantenimiento no sostenido e inadecuado. En función del problema a atacar, estos mantenimientos serán correctivos o perfectivos, si solventan fallos o incluyen mejoras que dan lugar a nuevos requisitos (como las mejoras planteadas en el apartado 5.2.1), respectivamente.

Se pueden dar diferentes modelos de CdV, pero el más recomendable, y que se ha seguido en el trabajo, es el modelo iterativo e incremental. Éste divide el software en fragmentos de mejor manejo y, así, aislar los fallos o secciones en desarrollo de los contenidos que no deberían sufrir modificaciones.

En el supuesto de la implantación de esta herramienta en la Escuela Naval, se plantea una solución para el mantenimiento de la aplicación llevado a cabo en la propia estructura de la organización en el Anexo II: Reflexiones Éticas y Sociales, con sus consideraciones en el ámbito social.

5.2.3 Explotación del proyecto actual en el ámbito académico

La línea futura en la visualización número 3 define la posibilidad de que la herramienta se incorpore a la enseñanza de formación de oficiales como un simulador fiel a la operación de las plataformas. El caso de la inmersión con realidad virtual no es una opción en el proyecto actual, pero sí se podrán simular otros tipos de situaciones. A partir de conjuntos de datos con fallos introducidos por el personal evaluador, y de un plan de reacción en cada tipo de fallo, el alumnado podrá realizar tareas de validación con los modelos del estado de equipos, y también de realizar apreciaciones sobre las acciones correctivas a tomar.

5.3 Líneas industriales

Existe otra rama de explotación de este proyecto en el ámbito del mantenimiento predictivo. La herramienta del detector de anomalías, disponible en la ventana de "Data" (ver apartado 3.6.1.2), permite de una forma tanto numérica como gráfica observar variaciones a los modelos de comportamiento habituales de los buques. Pretende por tanto ser una herramienta válida para llevar a cabo detecciones tempranas de estas anomalías, que puedan materializarse con el tiempo en fallos mayores.

Los modelos generados serán únicamente aplicables a la unidad y en las variables exactas seleccionadas en el caso de aquellas plataformas que se encuentran en servicio. Esto se debe a que se enfrentan a la problemática de no disponer de información sobre la línea base de los equipos y sistemas, para comprobación de discrepancias con las mediciones, por lo que no se parte de un conocimiento completo del rendimiento del material. Una línea industrial de desarrollo es el establecimiento de esta línea base, un trabajo de análisis e investigación en los historiales de los equipos para construir estos modelos, e incluso aplicar una tercera dimensión a los análisis de dos variables realizados, la del tiempo de servicio.

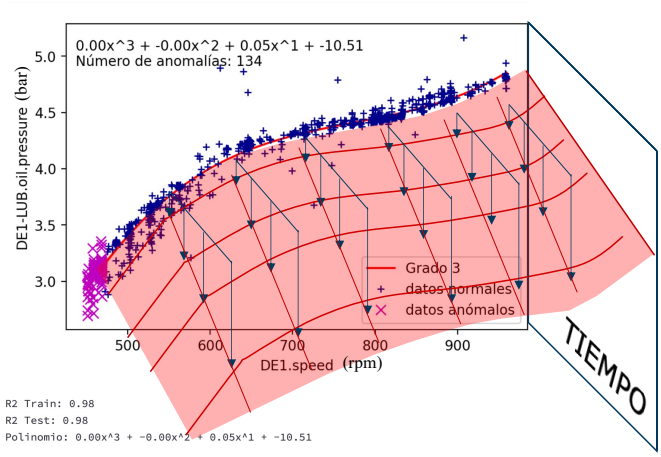


Figura 5-2. Modelo con variable temporal
Elaboración propia

La Figura 5-2 muestra un ejemplo del resultado de combinar el modelo generado por la aplicación para la presión del aceite de lubricación frente a las revoluciones del motor (modelo polinómico de tercer grado) con una disminución de esta presión con el avance del tiempo. Es otro factor a introducir en el visualizador de predicciones y podrá generar predicciones a largo plazo en función de los datos registrados.

5.4 Líneas específicas del Ministerio de Defensa

Por último, la Estrategia de Tecnología e Innovación para la Defensa (ETID), aprobada por la Secretaria de Estado de Defensa en el año 2020, la 11ª línea de acción en la que se pretende incidir son las tecnologías de la información, comunicaciones y simulación, Dentro de esta rama, el punto 11.3.2 pretende incorporar la simulación como herramienta para la toma de decisiones, como se menciona también en la tercera línea futura en el campo de visualización inmersiva.

También en la ETID, su Anexo A recoge en sus objetivos tecnológicos el aprovechamiento del avance tecnológico actual por medio de tecnologías para el mantenimiento predictivo de las plataformas de defensa. Como facilitadores, busca la integración de los métodos de Big Data, IoT, Inteligencia Artificial y visualización analítica, líneas que están recogidas en este proyecto [50].

6 BIBLIOGRAFÍA

[1] *Universo 4.0. impacto en las organizaciones.* [Película]. Armada Española, 2018.

[2] Estado Mayor de la Defensa, PDC-01(A) "Doctrina para el empleo de las Fuerzas Armadas", Madrid: Estado Mayor de la Defensa, 2018.

[3] G. C. Thorpe y S. L. Falk, «Pure Logistics: The Science of War preparation,» Washinton, NDU Press, 1986.

[4] Estado Mayor de la Armada, Concepto de Apoyo Logístico, 2017.

[5] UVICOA, Dirección, *Curso de Operador del Sistema Integrado de Control de Plataforma en fragatas de la clase Álvaro de Bazán (1ª serie).* [Película]. Centro de Ayudas a la Enseñanza de la Armada.

[6] J. D. Gonzalez Aller Lacalle y F. Lamas Lopez, «EVOLUCIÓN DEL CENTRO DE SUPERVISIÓN Y ANÁLISIS DE DATOS DE LA ARMADA (CESADAR),» *Revista General de Marina,* pp. 333-346, 2018.

[7] Órgano Auxiliar de la Jefatura del Apoyo Logístico (OAJ-JAL), Norma permanente de sostenimiento NUM 01/2022, del Almirante Jefe del Apoyo Logístico, sobre la organización, procedimientos y responsabilidades de usuarios del sistema Automatizado de Tareas de Vigilancia y Análisis de mantenimiento de la Armada (ATAVIA), Ministerio de Defensa. Armada, 2022.

[8] INDRA, Introducción a SOPRENE, Dirección General de Armamento y Material, 2020.

[9] C. Soriano González, Puesta en marcha virtual (virtual commissioning) en el contexto de la industria 4.0, Madrid: Universidad Politécnica de Madrid, ETSII, 2018.

[10] Navantia, Plan Estratégico de Navantia, 2018.

[11] M. Grieves y J. Vickers, «Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems,» de *Transdisciplinary perspectives on complex systems. New findings and approaches.*, Springer, 2017.

[12] Jefatura de Apoyo Logístico de la Armada, Visión del AJAL del Arsenal Inteligente, Armada, 2023.

[13] Y. Yoshimura, Mathematical Model for Manoeuvring Ship Motion (MMG Model), Hokkaido, Japon: School of Fisheries Sciences, 2005.

[14] D. Orive, A. López, E. Estévez, A. Orive y M. Marcos, «Desarrollo de gemelos digitales para la simulación e integración de activos de fabricación en la industria 4.0,» de *XLII Jornadas de Automática*, Castellón, 2021.

[15] Epic Games, «Twinmotion,» 2023. [En línea]. Available: https://www.twinmotion.com/features. [Último acceso: 2 Febrero 2023].

[16] Indra, Inteligencia Artificial enfocada al mantenimiento predictivo, INDRA sistemas, 2019.

[17] Universidad Politécnica de Cataluña, «Lógica difusa, la brillante ciencia de los grises,» *SINC,* 2009.

[18] P. G. González, Gemelo digital. Concepto y aplicación en la Armada Española, Universidad Politécnica de Madrid, 2018.

[19] G. Rodríguez Casajús, «Predicción de tipo de buque utilizando datos AIS y técnicas de inteligencia artificial,» 2022. [En línea]. Available: http://calderon.cud.uvigo.es/bitstream/handle/123456789/530/Rodr%C3%ADguez%20Casajús%2c%20Gonzalo%20-%20Memoria.pdf?sequence=1&isAllowed=y.

[20] J. W. Nilsson y S. A. Riedel, Circuitos eléctricos, Madrid: Pearson Educación, 2005.

[21] D. H. Maulud and A. M. Abdulazeez, "A Review on Linear Regression Comprehensive in Machine Learning," *Journal of Applied Science and Technology Trends ,* pp. 140-147, 31 Diciembre 2020.

[22] N. S. Narahari, J. Manasa y R. Gupta, «Machine Learning based Predicting House Prices using Regression Techniques,» de *2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, Bangalore, India, 2020.

[23] B. Manesh, «Machine Learning Algorithms - A Review,» *International Journal of Science and Research (IJSR),* vol. 9, 2020.

[24] G.-R. Ji y Y.-J. Z. Pu Han, «Wind speed forecasting based on support vector machine with forecasting error estimation,» de *Sixth International Conference on Machine Learning and Cybernetics*, Hong Kong, 2007.

[25] A. Navada, A. N. Ansari, S. Patil y S. Patil, «Overview of Use of Decision Tree algorithms in Machine Learning,» de *IEEE Control and System Graduate Research Colloquium*, Pune, India, 2011.

[26] S. E.R., «Understand Random Forest Algorithms With Examples,» *Analytics Vidhaya,* 15 Febrero 2023.

[27] IBM, «Redes neuronales,» [En línea]. Available: https://www.ibm.com/es-es/cloud/learn/neural-networks. [Último acceso: 15 Febrero 2023].

[28] S.-H. Han, K. W. Kim, S. Kim and Y. C. Youn, "Artificial Neural Network: Understanding the Basic Concepts without Mathematics," *Dement Neurocogn Disord,* pp. 83-89, 17 Septiembre 2018.

[29] E. Kavlakoglu, "AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What's the Difference?," 27 Mayo 2020. [Online]. Available: https://www.ibm.com/cloud/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks. [Accessed 1 Febrero 2022].

[30] Escuela Naval Militar, Pub. ENM 4810. Manual de las lanchas de instrucción A-121, Marín: Dirección de Enseñanza Naval.Armada Española, 2016.

[31] CATERPILLAR, Marine engine electronic displays installation guide, 2008.

[32] CUD-ENM, «Proyectos/Contratos,» [En línea]. Available: https://cud.uvigo.es/proyectos-contratos/.

[33] A. P. Seoane, El gemelo digital como base tecnológica para el desarrollo logístico de la Armada 4.0, X. Núñez Nieto, Ed., Marín, Pontevedra: CUD-ENM, 2020.

[34] Ghenova, «Gemelo digital-defensa,» 2020. [En línea]. Available: https://ghenova.com. [Último acceso: 13 Febrero 2023].

[35] International Baccaleurate Organization, A Guide to Origin, Purpose, Value and Limitations (OPVL), IBO, 2023.

[36] Arun Kumar y Supriya.P.Panda, « A Survey: How Python Pitches in IT-World,» de *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing*, Faridabad, India, 2019.

[37] P. G. Salgado, «Python,» 28 Febrero 2023. [En línea]. Available: https://docs.python.org/es/3.11/whatsnew/3.11.html#whatsnew311-pep654.

[38] C. Bellman, A. Seet y O. Baysal, «Studying Developer Build Issues and Debugger Usage via Timeline Analysis in Visual Studio IDE,» de *ACM/IEEE 15th International Conference on Mining Software Repositories*, Gothenburg, Sweden, 2018.

[39] J. K. Rask, F. P. Madsen, H. D. Macedo and N. Battle, "Visual Studio Code VDM Support," in *The 18th Overture Workshop*, Aarhus N, Denmark, 2020.

[40] IONOS, «Jupyter Notebook: documentos web para análisis de datos, código en vivo y mucho más,» 28 Febrero 2019. [En línea]. Available: https://www.ionos.es/digitalguide/paginas-web/desarrollo-web/jupyter-notebook/. [Último acceso: 16 Febrero 2023].

[41] NumFOCUS, "Pandas," OVHCloud, 2023. [Online]. Available: https://pandas.pydata.org/about/.

[42] Streamlit Inc., «Streamlit library,» 2023. [En línea]. Available: https://docs.streamlit.io/library.

[43] I. Stančin and A. Jović, "An overview and comparison of free Python libraries for data mining and big data analysis," in *MIPRO*, Opatija , 2019.

[44] Dirección General de Armamento y Material, Publicación Española de Calidad. Aseguramiento Oficial de la Calidad mutuo en la OTAN, 2015.

[45] Comité Técnico CTN 307 Gestión de Riesgos, UNE-ISO 31000, Madrid, 2018.

[46] P. Morales y L. Rodríguez, Application of the Kendall correlation and Spearman coefficients, Barquisimeto, Venezuela: Universidad Centroccidental "Lisandro Alvarado", 2016.

[47] M. B. Rodríguez, Ingeniería inversa aplicada: metodología y aplicaciones industriales, Medellín, Colombia: Universidad EAFIT, 2011.

[48] A. Carpentier y J. Eiras, Guía para el tratamiento de incertidumbres, CUD-ENM, 2022.

[49] International Organization for Standardization, *Norma ISO 25010. System and software quality models,* 2011.

[50] Ministerio de Defensa, Estrategia de Tecnología e Innovación para la Defensa, Madrid: Secretaría General Técnica, 2020.

[51] IMO, «International Maritime Organization,» 2019. [En línea]. Available: https://www.imo.org. [Último acceso: 6 Marzo 2023].

[52] C. Nuchturee, «A review of energy efficient methods for all-electric ships,» de *IOP Earth and Environmental Science*, 2018.

[53] American Bureau of Shipping, Ship Energy Efficiency Measures-status and guidance, ABS.

[54] World Economic Forum, Data Excellence: Transforming manufacturing and supply systems, 2021.

[55] World Economic Forum, The Data-Driven Journey Towards Manufacturing Excellence, 2022.

[56] P. Fraga-Lamas, D. Noceda-Davila, T. M. Fernández-Caramés, M. A. Díaz-Bouza y M. Vilar-Montesinos, «Smart Pipe System for a Shipyard 4.0,» *Sensors,* 2016.

[57] M. Cebral-Fernandez, D. Crespo-Pereira, A. Garcia-del-Valle y M. Rouco-Couzo, Improving planning and resource utilization of a shipbuilding process based on simulation, Ferrol, 2016.

[58] M. Rouco-Couzo, D. Crespo-Pereira, A. Garcia-del-Valle, M. Cebral-Fernandez y R. Rios-Prado, Discrete event simulation to improve cutting-welding workshop in a shipyard, 2016.

[59] International Labour Office, Sectoral Policies Department and International Labour Standards Department, Guidelines for implementing the occupational safety and health provisions of the Maritime Labour Convention, Génova, 2006.

[60] J. Rodríguez y V. Fernández, Cómo redactar el estado del arte de un trabajo, Editorial Genios, 2010.

[61] S. F. González, Aproximación metodológica a la implantación del gemelo digital en buques, V. Diaz Casas y M. Miguez Gonzalez, Edits., Tesis Doctoral UDC, 2022.

[62] W. Hu, T. Z. a. X. Deng and Z. L. a. J. Tan, "Digital twin: a state-of-the-art review of its enabling technologies, applications and challenges," 2020.

ANEXO I: IMPLICACIONES SOCIALES, ECONÓMICAS Y AMBIENTALES

En este anexo se justifica la consideración intrínseca de condicionantes de carácter no técnico en el desarrollo de este trabajo. Se realiza una aproximación desde una perspectiva conceptual, apoyándose en la literatura existente que justifica que la obtención dinámica de datos y su procesado en tiempo real constituye una metodología que permite mejorar aspectos medioambientales, económicos, industriales, sociales y de seguridad.

ASPECTOS MEDIOAMBIENTALES y ECONOMICOS.

Se consideran de forma conjunta estos dos aspectos porque se puede entender que, en el dominio marítimo, están relacionados.

La emisión de gases relacionados con el cambio climático presenta, en la industria marítima, una relación directa con el consumo de combustible. De la misma forma, el coste de operación de un barco también está directamente relacionado con el consumo de combustible. Por lo tanto, un factor fundamental para reducir tanto la emisión de gases de efecto invernadero como los costes durante la operación del buque es la reducción del consumo de combustible optimizando el funcionamiento de los motores propulsores y auxiliares, evitando parámetros anómalos.[12].

Se pone entonces de relieve la importancia de la eficiencia energética, entendida como " la reducción de consumo de energía, manteniendo los mismos servicios energéticos, sin disminuir el confort ni la calidad de vida, asegurando el abastecimiento de energía, protegiendo el medio ambiente y fomentando la sostenibilidad"

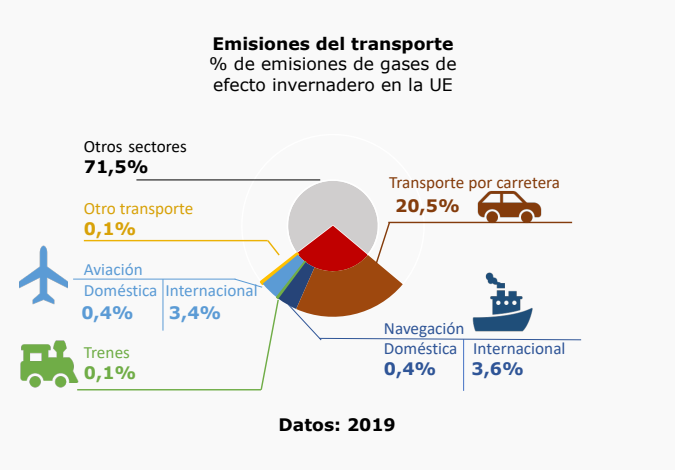


Figura Anexo I- 1. Emisiones del transporte en la UE.
Elaboración propia con datos de la Agencia Europea de Medio Ambiente

La normativa sobre eficiencia energética en el entorno marino.

[12] En la fase de diseño de los buques se pueden implementar otras medidas como el uso de combustibles como el Gas Natural Licuado o el hidrógeno, la optimización de las formas del buque para disminuir su resistencia o mejoras en las hélices.

La Organización Marítima Internacional (OMI) es una agencia perteneciente a las Naciones Unidas cuyos cometidos son la garantía de seguridad en la navegación, así como reducir la contaminación debida al transporte marítimo.

Se trata de la máxima autoridad mundial en el establecimiento de normativa para la seguridad, la protección y el comportamiento ambiental que debe cumplir la comunidad internacional en el transporte marítimo, de una manera justa y eficaz.

Las disposiciones que se adoptan en el seno de la OMI comprenden todos los ámbitos del transporte marítimo internacional con el fin de garantizar que este sector tan importante continúe siendo seguro, ecológico, eficiente en términos energéticos y protegido.

Formando parte de la OMI se encuentra el Comité de Protección del Medio Marino (MPEC, por sus siglas en inglés). Este comité está facultado para examinar toda cuestión que sea competencia de la organización en relación con la prevención y contención de la contaminación del mar, ocasionada por los buques [51].

El MEPC dictó una serie de normas relativas a la mejora de la eficiencia energética que se incluyeron como Anexo VI del Convenio para la prevención de la polución marina (MARPOL). Dicho anexo fue aprobado en el año 1997. Desde entonces, la normativa ha estado sometida a un proceso de revisión por parte del MPEC.

El capítulo 4 del anexo VI contiene las regulaciones relativas a la eficiencia energética en buques. En el MPEC 62 (julio del 2011), se estableció la obligación de calcular el Índice de Eficiencia Energética del Proyecto (EEDI) para los buques de nueva construcción y el desarrollo del Plan de Gestión de la eficiencia energética del buque (SEEMP) para todos los buques.

De carácter no obligatorio, se implementó el Índice Operacional (EEOI) el cual evalúa de forma cuantitativa el efecto de las medidas operativas, como la reducción de velocidad o la optimización de la navegación, sobre el consumo de combustible.

Recientemente se han introducido dos nuevos indicadores, que han entrado en vigor a partir del 1 de enero del 2023:

- Para los buques en servicio, se definió el Indicador de Eficiencia Energética en buques existentes (EEXI).
- No tanto en el ámbito de la eficiencia energética sino en el medioambiental, pero con una clara relación con los anteriores, se definió el Indicador de Intensidad de Carbón (CII)

La OMI ha fijado como objetivos la reducción, considerando la media del transporte marítimo mundial, de por lo menos el 40 % de CO_2 en el 2030 y de un 70 % en el 2050. Por otra parte, ha establecido el objetivo de reducir la emisión total anual de Gases de Efecto Invernadero de por lo menos el 50 % en 2050. Para el cálculo de las reducciones se toman como referencia los valores del 2008.

En el ámbito de los buques de guerra no existe una normativa internacional que aplique. La adhesión a las regulaciones contenidas en el MARPOL es de carácter voluntario por lo que depende de las políticas aplicadas en cada país. En el caso de la Armada Española, existe una voluntad de cumplir con lo establecido en la normativa de la OMI en la medida que sea compatible con los requisitos operativos de los buques [52].

Indicadores de eficiencia energética.

El EEDI

El cálculo del EEDI se realiza mediante una fórmula compleja definida en la Resolución MEPC.245(66). De forma simplificada, el EEDI se puede representar por la ecuación (23), cuyas unidades son gramos de dióxido de carbono por milla navegada:

$$EEDI = \frac{EMP + EMA + E_{PTI-AE} - E_{ME}}{VTR} \quad (23)$$

Donde:

- EMP: Emisiones de la propulsión principal que representa la energía total demandada por los motores principales
- EMA: Emisiones de motores auxiliares que representan la demanda eléctrica total de los consumidores del buque.
- E_{PTI-AE}: Emisiones que representan la energía en el eje de los motores/generadores que puede ser reducida por tecnología de ahorro de energía presentes en el buque
- E_{ME}: Reducción de las emisiones de la propulsión principal debido al empleo de tecnologías propulsivas de ahorro de energía (como velas o cometas)
- VTR: Valor del trabajo realizado

El SEEMP

El plan de gestión de la eficiencia energética del buque SEEMP es una medida operativa que establece un mecanismo que permite mejorar la eficiencia energética de un buque, de manera rentable, aplicando cambios operativos como el planeamiento de cada viaje considerando las condiciones meteorológicas, la optimización de la velocidad, la potencia en el eje, el trimado, el lastre, etc.

El EEOI

Las directrices para el uso cálculo voluntario del EEOI se establecen en el documento MPEC.1/Circ.694. El EEOI mide el consumo eficiente del buque y de forma cuantitativa evalúa los valores de las toneladas de dióxido de carbono emitido en relación con las toneladas de carga por milla navegada.

Así, para un periodo determinado en el que se desarrollan varios viajes, el EEOI se obtendría mediante la ecuación (24):

$$EEOI_{Average} = \frac{\sum_j \sum_i FC_{ij} \times C_{Fj}}{\sum_i m_{carga} \times D_I} \quad (24)$$

Donde:

- *j*: es el tipo de combustible empleado en la navegación
- *i*: es el número de viaje al que corresponde el consumo
- FC_{ij}: es la masa de combustible *j* consumido en el viaje *i*.
- C_{Fj}: es el factor de conversión de la masa de combustible j en masa de CO_2. Se proporciona en el punto 3 de la Circular 694
- $m_{carga\ i}$: es la carga transportada en Tn o el trabajo realizado (número de TEU[13] o pasajeros)
- D_i: es la distancia recorrida en el viaje *i* en millas náuticas.

[13] TEU: Twenty foot Equivalent Unit. Contenedor normalizado de 20 pies de longitud

El EEXI y el CII

A partir del 1 de enero del 2023 es obligatorio que todos los buques calculen su EEXI e informen de su CII.

El EEXI es un indicador similar al EEDI, pero aplicable a buques que ya estaban en servicio cuando entró en vigor dicho indicador por lo que en su diseño no se tuvo en cuenta.

El **EEXI alcanzado** por un buque indica su eficiencia energética en comparación con una línea de base, que se denomina **EEIX requerido**. En caso de que un buque alcance el EEXI, éste será comparado con un índice de reducción de eficiencia energética válido para buques existentes. El valor del EEXI requerido no debe superar al calculado para un buque, garantizando así el cumplimiento de una normativa mínima de eficiencia energética

El CII determina la reducción anual necesaria para garantizar la mejora continua de la intensidad de carbono operacional de un buque dentro de un nivel de clasificación concreto. El CII operacional anual real alcanzado deberá compararse con adecuación al valor equivalente de CII prescrito. Así se determina la clasificación de la intensidad de carbono operacional.

En función del indicador de intensidad de carbono de un buque, su intensidad de carbono se calificará como A, B, C, D o E (siendo A la mejor). La calificación indica un nivel de rendimiento muy superior, superior, moderado, inferior o muy inferior. El grado de rendimiento se registrará en una "declaración de cumplimiento" que se detallará en el Plan de gestión de la eficiencia energética del buque (SEEMP).

Un buque calificado con D durante tres años consecutivos, o E durante un año, tendrá que presentar un plan de medidas correctivas para mostrar cómo se alcanzará el índice requerido de C o superior. Aquellos buques clasificados como A o B son sujeto de beneficiarse de los incentivos que las administraciones, autoridades portuarias y otras partes interesadas puedan conceder [53].

Optimización de la eficiencia energética mediante metodologías basadas en la obtención datos, su procesado y el desarrollo de modelos

Los indicadores descritos en el punto anterior reflejan la importancia que la comunidad marítima le da al medioambiente, estableciendo un objetivo claro de optimizar la eficiencia energética de los sistemas a bordo. Con este objetivo se persigue reducir el consumo de combustible, de forma que se reduzca la emisión de gases nocivos a la atmósfera, tenido además un efecto económico al reducir los costes de operación del buque.

En el desarrollo de este punto se identifican diversos estudios que demuestran que las metodologías basadas en la obtención y el procesado de datos son habilitadoras para optimizar la eficiencia energética de los buques, reducir el consumo de combustible y, como consecuencia, reducir la emisión de gases nocivos a la atmósfera y a reducir el coste de operación de los buques.

Estudio: "Data driven energy efficiency of ships". Tarik Taspinat. Master´s Thesis, Halmstad University.

Plantea la gran influencia que tiene la velocidad del buque en el consumo de combustible. En una aproximación simplista, se considera que una disminución de la velocidad en un 10% lleva aparejada una reducción en la energía demandada para la propulsión de un 27%. De forma contraria a esta percepción popular, algunos armadores han expresado que el consumo de combustible puede incrementar incluso si la velocidad se disminuye debido a un funcionamiento no eficiente de los motores. El estudio tiene como objetivo la optimización de la velocidad del buque, desarrollando modelos predictivos basados en los siguientes datos:

- Velocidad sobre el fondo

- Consumo
- Lastre
- Capacidad de los tanques de combustible
- Potencia del generador auxiliar
- Capacidad de carga
- Capacidad de los tanques de agua
- Rumbo
- Timado
- Velocidad del viento
- Dirección del viento

Estudio: "Data-driven approach for estimating power and fuel consumption of ships: a case of container vessel. Tayfun Uyanık , Yunus Yalman , Özcan Kalenderli , Yasin Arslanoglu , Yacine Terriche ,Chun-Lien Su and Josep M. Guerrero.

Establece que los componentes críticos empleados en los cálculos de la eficiencia energética de los buques son la potencia en el eje y el consumo de combustible. Debido a una interpretación no correcta de estos valores, se producen errores en el cálculo de los índices de eficiencia energética. El estudio tiene como objetivo el predecir la potencia en el eje y el consumo de combustible de un buque contenedor empleando metodologías basadas en datos que emplean los datos reales dinámicos del viaje para apoyar el proceso de cálculo de los índices de eficiencia energética del buque.

Estudio: Operational data-driven energy performance assessment of ships. The case study of a naval vessel with hybrid propulsion. Vasilikis, N. I.; Geertsma, R. D.; Visser, K.Journal of Marine Egineering and Technology

Introduce una metodología para la evaluación del desempeño energético del buque empleando datos procedentes del sistema integrado de control de plataforma. El buque objeto del estudio es un patrullero oceánico de la clase Holland.

Se trabaja con los datos siguientes:

- Velocidad del motor principal
- Consumo del motor principal
- Velocidad de los diésel generadores
- Potencia de los diésel generadores
- Velocidad del eje propulsor
- Par del eje propulsor
- Paso de la hélice
- Velocidad sobre el fondo

El estudio:

- Presenta un método para enriquecer el paquete de datos operativos, en caso de ausencia de parámetros clave, empleando modelos consolidados.
- Usa dicho paquete de datos enriquecido para evaluar el desempeño energético del buque considerando todo el rango operativo, en lugar de hacerlo en un número limitado de puntos.
- Introduce indicadores adecuados del desempeño energético del buque no sólo para evaluar operaciones pasadas del buque, sino también para apoyar la mejora de operaciones futuras.

- Examina el impacto de decisiones de diseño y operativas sobre el desempeño energético de buques equipados con propulsión híbrida (combinada diésel-eléctrica)

Estudio: Data-Driven Ship Digital Twin for estimating the speed loss caused by the marine fouling. Andrea Coraddua, Luca Onetob, Francesco Baldic, Francesca Cipollinib,Mehmet Atlara, Stefano Savio

Desarrolla un modelo basado en los datos obtenidos de los sensores de monitorización instalados a bordo de los buques para comparar la predicción de velocidad con la velocidad real cuya diferencia representa la pérdida de velocidad del buque.

Demuestra que la variación media de la pérdida de velocidad en función del tiempo permite estimar los efectos, de forma precisa y efectiva, del crecimiento de organismos marinos en el casco y las hélices, lo que permite programar de forma más eficiente la limpieza de estos.

Los datos capturados de los sensores del buque son:

- Situación (latitud y longitud)
- Consumo de los motores principales
- Consumo motores auxiliares
- Potencia en el eje
- Revoluciones de la hélice
- Calados a proa y popa
- Calados a estribor y a babor
- Velocidad relativa del viento
- Dirección relativa del viento
- Rumbo GPS
- Velocidad sobre el fondo
- Velocidad sobre el agua
- Sonda
- Temperatura del agua
- Densidad y temperatura del combustible
- Presión ambiental
- Humedad y punto de rocío
- Par del eje
- Ángulo de timón
- Aceleraciones
- Ángulos de cabezada, guiñada y balance

Aspectos industriales

En el ámbito industrial, la metodología desarrollada en este trabajo presenta grandes ventajas y está representada en el máximo nivel de complejidad por los Gemelos Digitales de los procesos, los equipos/sistemas e incluso de la fábrica.

En un ámbito global, el Foro Mundial de Economía (WEF) , en su informe "*Data Excellence: Transforming manufacturing and supply systems*", establece que el valor que proporciona la obtención de datos, su procesado y el desarrollo de modelos se pueda categorizar en tres grupos:

- Incremento de la productividad en los procesos de fabricación y suministro de sistemas
- Mayor satisfacción de los clientes a través de la mejora de productos y servicios.
- Impacto positivo en la sociedad (producción ética) y en el medioambiente (menor huella de carbono maximizando la eficiencia energética).

Dicho informe contempla que en una encuesta realizada en el 2020 por Boston Consulting Group a más de 1.300 empresas productivas, el 80 % de ellas considera que el alcanzar objetivos relacionados con la productividad como la motivación principal para establecer una metodología basada en datos [54].

Un informe posterior del WEF describe el trabajo realizado en el marco de una iniciativa lanzada en el 2019 que se denomina "*Unlocking Value in Manufacturing through Data Sharing".* En el documento se describen una serie de casos de uso reales en los que el empleo de metodologías basadas en datos constituye un factor de éxito, validando la importancia de estas en el ámbito industrial. A modo de ejemplo se identifican los siguientes casos:

- ***Ford Otosan***: Planeamiento de los procesos productivos y de las líneas productivas de las fábricas.
- ***ZF*:** Reducción de accidentes laborales un sus centros de distribución y fábricas.
- **Analog Devices**: Ejecución y control del proceso de fabricación automática.
- **Henkel**: Optimización avanzada de la ruta de distribución de mercancías.
- **Henkel**. Predicción precisa de demanda futura.
- **Arcelik**. Gestión inteligente de la energía.
- **Analog Devices**. Mejora de los productos mediante el desarrollo de modelos digitales de los mismos.

La industria de la construcción naval no es ajena a los beneficios productivos derivados de metodologías basadas en datos [55].

Como ejemplo de esta conciencia de las empresas del sector, en el año 2015 se constituye la Unidad Mixta de Investigación (UMI) entre Navantia y la Universidad da Coruña (UDC) que nace de la inquietud de la empresa constructora por iniciar un profundo proceso de transformación orientado a la sostenibilidad de la empresa en el mercado del siglo XXI. La UMI tiene por objetivo avanzar en la mejora tecnológica de un astillero que desarrolla productos de clase mundial. Fundamentalmente, los objetivos se centraron en mejorar la productividad de los procesos la competitividad frente a sus competidores.

Los trabajos científicos desarrollados por la UMI han sido prolíficos. Algunos de estos estudios desarrollaban metodologías basadas en modelos generados a partir de datos. A modo de ejemplo, si listan algunos de ellos a continuación:

Estudio: "Smart Pipe System for a Shipyard 4.0".

Se presenta una metodología basada en los datos dinámicos transmitidos de las tuberías que tienen que ser instaladas a bordo, de forma que, después de procesarlos, permiten su explotación (trazabilidad, localización, análisis del proceso, avisos, etc…) a través de diversas aplicaciones. [56]

Estudio: "Improving planning and resource utilization of a shipbuilding process based on simulation".

Los objetivos del trabajo eran el proporcionar una herramienta capaz de estimar los tiempos de fabricación del buque desde el corte inicial de chapa hasta la botadura, obtener una herramienta capaz de reproducir el efecto de diversa variables (productividad de los talleres, ritmo de soldadura, secuencia de bloques en grada, etc…) y diseñar un modelo parametrizado y flexible capaz de simular el proceso constructivo de cualquier barco. Los datos de entrada para el proceso de modelado se obtienen de hojas de cálculo de MS Excel [57].

Estudio: "Discrete event simulation to improve cutting-welding workshop in a shipyard"

Este trabajo describe un el desarrollo y validación de un modelo de simulación , basado en datos registrados en un taller de corte y soldadura de un astillero [58].

En el ámbito de la operación y el sostenimiento de los buques, el empleo de modelos basados en datos permite detectar funcionamientos anómalos de los sistemas, equipos y componentes constituyendo una herramienta del mantenimiento predictivo. Este es el caso del sistema ATAVIA, descrito en el punto 2.3.1 de este trabajo.

Aspectos sociales

El empleo de metodologías basadas en la obtención de datos permite implementar mejoras de ámbito social dentro del marco de la comunidad marítima. Un ejemplo lo constituye la mejora de las condiciones de habitabilidad y confort a bordo de los buques, optimizando la regulación de los sistemas de calefacción, ventilación y aire acondicionado (HVAC).

En el 61 Congreso de Ingeniería Naval e Industria Marítima (Palma de Mallorca, 2022), se presentó una ponencia con el título "Sistemas HVAC verdes y eficientes" en la que se exponía como mediante la obtención de datos (del nivel ocupacional, de parámetros de la calidad del aire interior y de las condiciones ambientales exteriores), su procesado y modelado se podían mejorar las condiciones de habitabilidad y confort a bordo.

Aspectos de seguridad y salud en el trabajo

De acuerdo con lo establecido por la *"International Labour Organization",* los riesgos relativos a la seguridad y salud del personal que desarrolla su trabajo a bordo de los buques son:

- Exposición al ruido
- Exposición a las vibraciones
- Iluminación artificial
- Radiación no ionizante
- Temperaturas extremas
- Trabajos en espacios confinados
- Peligros asociados al uso de equipos y maquinaria.

Las metodologías de desarrollo de modelos basados en datos permiten detectar anomalías de forma que se asegure el correcto mantenimiento de los equipos y el uso óptimo de los mismos lo que permitirá que dichos equipos trabajen alrededor del punto de diseño reduciendo su nivel de ruidos y vibraciones [59].

.

ANEXO II: REFLEXIONES ÉTICAS Y SOCIALES

El presente anexo muestra las reflexiones realizadas por el autor durante el desarrollo del trabajo en el plano ético y social. Por una parte, se incluye la cuestión de la propiedad del dato, referido a la definición de un dueño de la información recopilada y analizada. Por otro, la involucración de personal para asegurar el ciclo de vida del software, lo que implicaría una redistribución de la Plantilla orgánica (PLOR) de la UCO de explotación, o incluso la transferencia de esta tarea a UCOs especializadas.

Propiedad del dato

En primer lugar, el tratamiento de la información para labores de mantenimiento y toma de acciones correctivas suele ser una tarea auxiliada por agentes externos a las Fuerzas Armadas, por medio de contratos con el MINISDEF. Esto, a simple vista, puede parecer no suponer un problema, ya que la empresa civil, en el caso de la industria nacional de defensa, tiene también toda la información de las capacidades y limitaciones ("Caps & Lims") de los sistemas de armas por ser los principales contratistas. Sin embargo, con la entrega de estos sistemas a cada uno de los ejércitos, el sostenimiento de las unidades pasa a ser un condicionante operativo, que debe estar sujeto a las más altas clasificaciones de seguridad. La filtración de una variación operativa (VAROP) en las capacidades de la plataforma puede suponer una amenaza por medio de la explotación de esta asimetría en las líneas de acción del oponente.

¿A quién pertenecen los datos que generan los buques?

Esta reflexión se deberá realizar desde la concienciación por la seguridad de la información (SEGINFO), debiendo ser el responsable aquel órgano que sea capaz de aprovechar su explotación sin descuidar su reserva.



Figura Anexo II- 1. Diagrama de capacidades de los órganos de sostenimiento
Elaboración propia

La Figura Anexo II- 1 muestra una representación del grado de seguridad que los distintos organismos pueden ofrecer a los datos manipulados, así como el apoyo que se puede ofrecer a las unidades.

Ciclo de Vida del Software, implicaciones en la plantilla de la Armada

De acuerdo con lo expuesto en el apartado 5.2.2, el desarrollo de cualquier programa está sujeto a su mantenimiento durante las fases de todo su ciclo de vida. El ciclo de vida del software, como el de un producto, comprende las fases de concepción, desarrollo, producción, explotación, apoyo y retirada. En

todas ellas, con una mayor importancia en las etapas finales tiene una labor relevante el mantenimiento ofrecido [44]. Como cualquier tarea, se necesitan unos recursos materiales, económicos y humanos, para llevarla a cabo. En el supuesto de que esta aplicación de vigilancia y análisis fuese adoptada por la Armada, esto llevará a un incremento de la carga de trabajo del personal. Se plantean a continuación algunas opciones:

- En el caso de que su uso se limitase a las LLII de la Escuela Naval Militar: el personal responsable de su mantenimiento podría ser la dotación del núcleo de lanchas, con apoyo de la jefatura de mantenimiento de la UCO. Al no ser puestos que por PLOR requieran personal especializado en trabajos informáticos, sería necesaria la creación de un cursos que transfiriese los conocimientos mínimos para este personal, con impacto en su formación y en su perfil de carrera.
- En el caso de ser considerada un software válido para su distribución a otras unidades, podrá designarse personal dentro de cada UCO como en el caso anterior, o transferir la responsabilidad mantenedora a organismos especializados, como el CESADAR (apartado 2.3). Si bien es cierto que, en la actualidad, la Armada cuenta con un elevado número de aplicaciones logísticas, con sus propias necesidades de mantenimiento. La adición de nuevos software generaría una carga de trabajo adicional, llevando a necesidades de ampliación de las plantillas. Con este escenario, uno de los rumbos más adecuado a tomar podría ser el desarrollo de una única aplicación que garantice la cobertura de las capacidades cubiertas por los software actuales. Llevando a cabo esta tarea, y con la gestión del sostenimiento al completo desde una única aplicación, se cumpliría la definición de Armada 4.0 que abre este trabajo en la introducción, la búsqueda de la optimización de la relación interdepartamental de la mima.

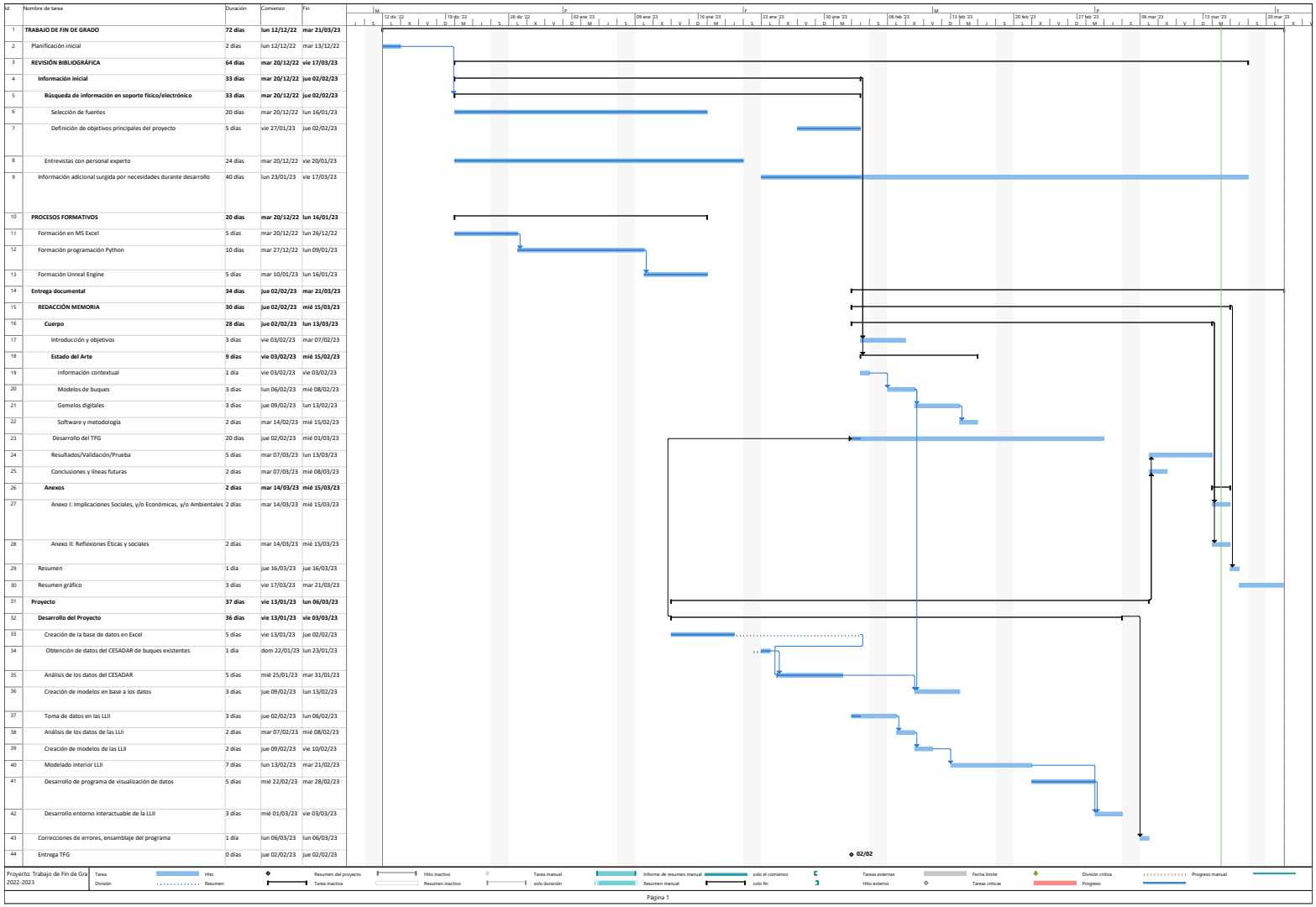
ANEXO III: GLOSARIO DE SIGLAS, ACRÓNIMOS Y ABREVIATURAS CONTENIDAS EN EL TRABAJO

AJEMA	Almirante Jefe del Estado Mayor de la Armada
ANN/SNN	Artificial Neural Network/ Simulated Neural Network
ARCA	Área del Repositorio Corporativo de la Armada
ATAVIA	Automatización de Tareas de Vigilancia y Análisis
BAC	Buque de Aprovisionamiento en Combate
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CdV	Ciclo de Vida
CESADAR	Centro de Supervisión y Análisis de Datos de la Armada
CFD	Computational Fluid Dynamics
CIA2	Centro de Inteligencia Artificial de la Armada
CSV	Comma Separated Values
DGAM	Dirección Generasl de Armamento y Material
DIC	Dirección de Ingeniería y Construcciones de la JAL
DIGEC	Dirección de Gestión Económica de la JAL
DIN	Dirección de Infraestructura de la JAL
DISOS	Dirección de Sostenimiento de la JAL
DoD	Departemnt of Defence (EEUU)
EMAD	Estado Mayor de la Defensa
EUR-ACE	European Accreditation of Engineering Programmes
FPV	First Person View
FUCOM	Fuerza de Combate de Superficie
GD(DT)	Gemelo digital (Digital Twin)
GUI	Graphical User Interface
HDR	High Dynamic Range
I3D	Infraestructura Integral de Información de Defensa
IDE	Integrated Development Environment
IoT	Internet of Things
JAL	Jefatura de Apoyo Logístico de la Armada
JUTAC	Junta de Táctica del Centro de Doctrina de la Flota
LIDAR	Light Detection and Ranging
LLII	Lanchas de Instrucción
LOBA	Listado Oficial de Buques de la Armada
MEVIMAN	Módulo Embarcado de Vigilancia y Mantenimiento
ML	Machine Learning
MMG	Mathematical Modelling Group
MPD	Marine Power Display
MS	Microsoft
MSE	Mean Square Error
PANDAS	Python for Data Analysis
PECAL	Publicación Española de Calidad

PLM	Product Lifecycle Management
PRFV	Poliéster Reforzado con Fibra de Vidrio
PTL	Procesos de Trabajo Logísticos
SICP	Sistema Integrado de Control de Plataforma
SIMNAV	Simulador de Navegación de la Escuela Naval Militar
SLI	Sistema Logístico Integrado
SMBC	Sistema de Mantenimiento Basado en la Condición
SOPRENE	Sostenimiento Predictivo Basado en Redes Neuronales
SVM	Support Vector Machines
TIC	Tecnologías de la Información y la Comunicación
UCO	Unidad, Centro u Organismo
UVICOA	Unidad de Enseñanza Virtual Corporativa de la Armada

ANEXO IV: PLANIFICACIÓN TEMPORAL DEL TRABAJO

La duración de este trabajo de fin de grado está comprendida entre la fecha 5 de Noviembre de 2022 y su entrega el 24 de Marzo de 2023. Para la ejecución de las diferentes etapas y tareas se ha realizado la siguiente planificación.



Id	Nombre de tarea	Duración	Comienzo	Fin
1	**TRABAJO DE FIN DE GRADO**	**72 días**	**lun 12/12/22**	**mar 21/03/23**
2	Planificación inicial	2 días	lun 12/12/22	mar 13/12/22
3	**REVISIÓN BIBLIOGRÁFICA**	**64 días**	**mar 20/12/22**	**vie 17/03/23**
4	**Información inicial**	**33 días**	**mar 20/12/22**	**jue 02/02/23**
5	**Búsqueda de información en soporte físico/electrónico**	**33 días**	**mar 20/12/22**	**jue 02/02/23**
6	Selección de fuentes	20 días	mar 20/12/22	lun 16/01/23
7	Definición de objetivos principales del proyecto	5 días	vie 27/01/23	jue 02/02/23
8	Entrevistas con personal experto	24 días	mar 20/12/22	vie 20/01/23
9	Información adicional surgida por necesidades durante desarrollo	40 días	lun 23/01/23	vie 17/03/23
10	**PROCESOS FORMATIVOS**	**20 días**	**mar 20/12/22**	**lun 16/01/23**
11	Formación en MS Excel	5 días	mar 20/12/22	lun 26/12/22
12	Formación programación Python	10 días	mar 27/12/22	lun 09/01/23
13	Formación Unreal Engine	5 días	mar 10/01/23	lun 16/01/23
14	**Entrega documental**	**34 días**	**jue 02/02/23**	**mar 21/03/23**
15	**REDACCIÓN MEMORIA**	**30 días**	**jue 02/02/23**	**mié 15/03/23**
16	**Cuerpo**	**28 días**	**jue 02/02/23**	**lun 13/03/23**
17	Introducción y objetivos	3 días	vie 03/02/23	mar 07/02/23
18	**Estado del Arte**	**9 días**	**vie 03/02/23**	**mié 15/02/23**
19	Información contextual	1 día	vie 03/02/23	vie 03/02/23
20	Modelos de buques	3 días	lun 06/02/23	mié 08/02/23
21	Gemelos digitales	3 días	jue 09/02/23	lun 13/02/23
22	Software y metodología	2 días	mar 14/02/23	mié 15/02/23
23	Desarrollo del TFG	20 días	jue 02/02/23	mié 01/03/23
24	Resultados/Validación/Prueba	5 días	mar 07/03/23	lun 13/03/23
25	Conclusiones y líneas futuras	2 días	mar 07/03/23	mié 08/03/23
26	**Anexos**	**2 días**	**mar 14/03/23**	**mié 15/03/23**
27	Anexo I: Implicaciones Sociales, y/o Económicas, y/o Ambientales	2 días	mar 14/03/23	mié 15/03/23
28	Anexo II: Reflexiones Éticas y sociales	2 días	mar 14/03/23	mié 15/03/23
29	Resumen	1 día	jue 16/03/23	jue 16/03/23
30	Resumen gráfico	3 días	vie 17/03/23	mar 21/03/23
31	**Proyecto**	**37 días**	**vie 13/01/23**	**lun 06/03/23**
32	**Desarrollo del Proyecto**	**36 días**	**vie 13/01/23**	**vie 03/03/23**
33	Creación de la base de datos en Excel	5 días	vie 13/01/23	jue 02/02/23
34	Obtención de datos del CESADAR de buques existentes	1 día	dom 22/01/23	lun 23/01/23
35	Análisis de los datos del CESADAR	5 días	mié 25/01/23	mar 31/01/23
36	Creación de modelos en base a los datos	3 días	jue 09/02/23	lun 13/02/23
37	Toma de datos en las LLII	3 días	jue 02/02/23	lun 06/02/23
38	Análisis de los datos de las LLII	2 días	mar 07/02/23	mié 08/02/23
39	Creación de modelos de las LLII	2 días	jue 09/02/23	vie 10/02/23
40	Modelado interior LLII	7 días	lun 13/02/23	mar 21/02/23
41	Desarrollo de programa de visualización de datos	5 días	mié 22/02/23	mar 28/02/23
42	Desarrollo entorno interactuable de la LLII	3 días	mié 01/03/23	vie 03/03/23
43	Correcciones de errores, ensamblaje del programa	1 día	lun 06/03/23	lun 06/03/23
44	Entrega TFG	0 días	jue 02/02/23	jue 02/02/23

8 **Entrevistas con personal experto**
-TN (CIA) José Costal. CESADAR Periférico Ferrol.
-CF (CGA) Touza Gil . CIA2
11 **Formación en MS Excel**
VBA lanquaje
12 **Formación programación Python**
VS Code // Jupyter Notebooks Extension
37 **Toma de datos en las LLII**
Embarque en LLII

Página 3

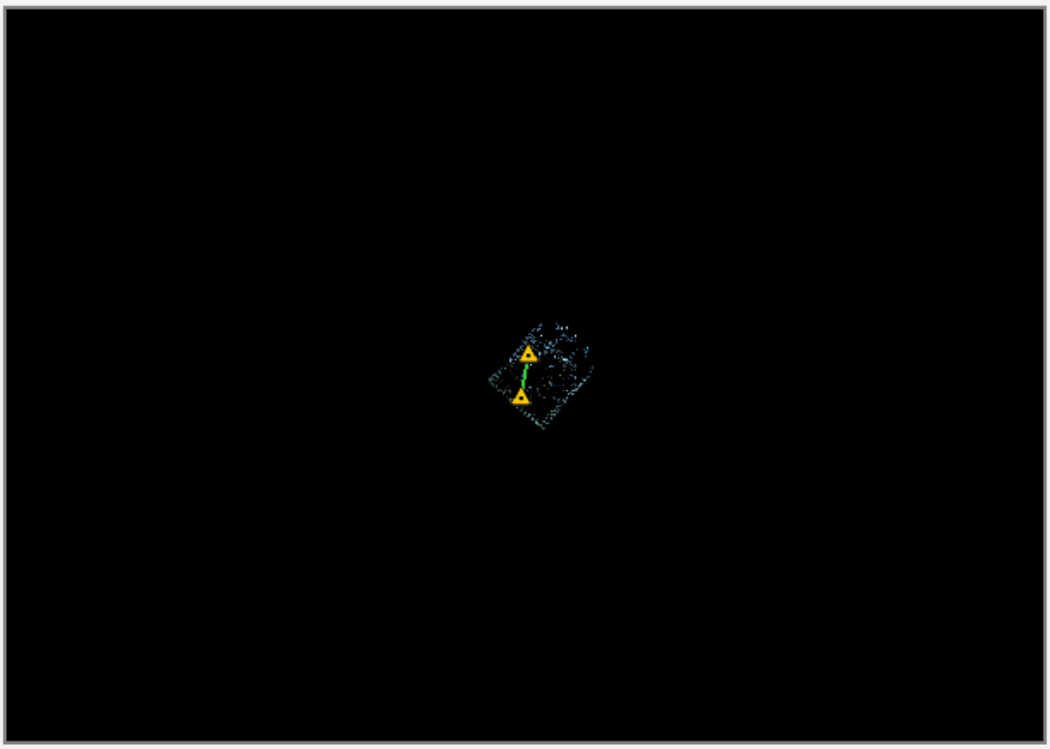
ANEXO V: INFORME LEICA CYCLONE REGISTER 360 PLUS

Cyclone REGISTER 360
Informe de registro

Leica Geosystems

feb. 28, 2023

Certificado por:



SiteMap 1

Calidad general

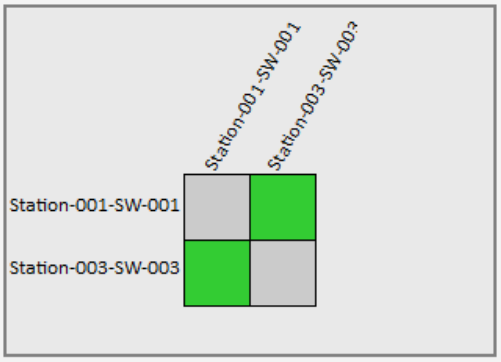
Resultados de error para Conjunto 1

Número de estacionamientos:	2
Número de enlaces:	1
Fuerza:	80 %
Solapamiento:	63 %

Error de conjunto 0.013 m ✓	
Solapamiento 63 % ✓	Fuerza 80 % ✓
Nube a nube 0.013 m ✓	Error de diana --

Error máximo de 0.015 m. Error máximo de 0.020 m. Error mayor de 0.020 m.

Matriz de calidad de enlaces (1 de 1) -



Resultados de error de enlace

1 Vista general

Nombre de enlace	Estacionamiento 1	Estacionamiento 2	Solapamiento	Error medio Abs.
Enlace 1	Station-001-SW-001	Station-003-SW-003	63 %	**0.013 m**

2 Detalles

Nombre de enlace	Estacionamiento 1	Estacionamiento 2	Solapamiento	Error medio Abs.
Enlace 1	Station-001-SW-001	Station-003-SW-003	63 %	**0.013 m**
		Nube a nube		**0.013 m**
		Diana	Error medio de diana:	**--**

ANEXO VI: CÓDIGO FUENTE PÁGINA HOME

```
import streamlit as st
from PIL import Image
import pandas as pd


#Configuración inicial de la página
st.set_page_config(page_title="Programa de seguimiento de las LLII de la ENM",
page_icon=None, layout="wide", initial_sidebar_state="auto")

#Selección del tema, con los colores marcados por el Manual de Imagen Corporativa del AJEMA
theme = """
[theme]
base="light"
primaryColor="#173763"
secondaryBackgroundColor="#c7c7c7"

"""
st.write(f'<style>{theme}</style>', unsafe_allow_html=True)




#Llamamiento a imagen superior dibujo de las LLII (elaboración propia)
image= Image.open(' /IMAGES/fondomenu.jpg')
st.image(image,
         width=1000
         )




#Llamamiento a las imáges de los logos de la ENM y Armada para incluir en la barra lateral
image = Image.open(' /IMAGES/escudo-enm.png')
st.sidebar.image(image, width=150)
image = Image.open(' /IMAGES/IMAGOTIPOARMADA.png')
st.sidebar.image(image, width=150)


#Título y subtítulo de la página
st.header('Ventana de visualización de datos')
st.subheader('AI de la serie Rodman 66')

excel_file = st.file_uploader("Selecciona un archivo Excel", type=["xlsx", "xls","xlsm"])

#En caso de que el usuario haya seleccionado un archivo formato "type", se guarda el doc
como xls
if excel_file is not None:


    xls = pd.ExcelFile(excel_file)
```

```
    #Con la librería pandas se obtienen los datos de las hojas que contiene el Excel
    sheet_names = xls.sheet_names

    # Adición de un multiselect para seleccionar la hoja que contiene los datos de la
plataforma
    selected_sheets = st.multiselect("Seleccione la hoja a analizar",
sheet_names,max_selections=1)


    if  len(selected_sheets)==1:

        #Cargar los datos del libro de Excel, almacenado como primer elemento del array
selected_sheets
        excel_file = f'{excel_file.name}'
        data_name = selected_sheets[0]
        df = pd.read_excel(excel_file, sheet_name=data_name, usecols='A:R', header=0)


        vel_inicial=0
        vel_final=45

        #Crear slider de doble fin limitado por las velocidades a analizar. Se eleigirán
velocidades entre los 0 y 45 nudos, este último por ser la máxima velocidad alcanzada en un
buque de guerra en la historia (Destructor "Le Terrible, 1935")
        minv_elec, maxv_elec = st.slider(
                    "Selecciona un rango de velocidades para comparar los datos",
                    value=(vel_inicial, vel_final),
                    min_value=vel_inicial,
                    max_value=vel_final)

        speed=df['SHIP.overground.speed']
        df = df.loc[(speed<=maxv_elec) & (speed>=minv_elec),:]

        spearman_corr = df.corr(method='spearman')
        st.write(spearman_corr.style.background_gradient(cmap='Blues'))
```

ANEXO VII: CÓDIGO FUENTE PÁGINA DATA

```
import pandas as pd
import streamlit as st
import seaborn as sns
import matplotlib.pyplot as plt
from PIL import Image
import datetime as dt
import numpy as np
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import PolynomialFeatures
from sklearn.metrics import r2_score
import plotly.express as px
from sklearn import svm


    # Configurar el tema de la página
    st.set_page_config(
        page_title="Programa de seguimiento de las LLII de la ENM",
        layout="wide",
        initial_sidebar_state="expanded",
    )

    # Definir el tema
    theme = """
    [theme]
    base="light"
    primaryColor="#173763"
    secondaryBackgroundColor="#c7c7c7"

    """

    # Aplicar el tema
    st.write(f'<style>{theme}</style>', unsafe_allow_html=True)

    # Imagen de logo de la ENM y Armada
```

```
    image = Image.open('/IMAGES/escudo-enm.png')
    st.sidebar.image(image, width=150)
    image = Image.open(' /IMAGES/IMAGOTIPOARMADA.png')
    st.sidebar.image(image, width=150)



    # Encabezados de la página
    st.header('Ventana de visualización de datos')
    st.subheader('AI de la serie Rodman 66')


    excel_file = st.file_uploader("Selecciona un archivo Excel", type=["xlsx",
"xls","xlsm"])


    # Si se seleccionó un archivo, guardamos su ruta
    if excel_file is not None:



        xls = pd.ExcelFile(excel_file)


        #Con la librería pandas se obtienen los datos de las hojas que contiene el Excel
        sheet_names = xls.sheet_names


        # Adición de un multiselect para seleccionar la hoja que contiene los datos de la
platafroma
        selected_sheets = st.multiselect("Seleccione la hoja a analizar",
sheet_names,max_selections=1)



        if  len(selected_sheets)==1:


            # Cargar los datos del archivo de Excel
            excel_file = f'{excel_file.name}'
            data_name = selected_sheets[0]


            df = pd.read_excel(excel_file, sheet_name=data_name, usecols='A:O', header=0)


            # Selector de opciones de visualización según las cabeceras de las columnas del
libro Excel y hoja seleccionados
```

```
            dfvalores = df.columns.values
            selection = st.multiselect('Valores', dfvalores, max_selections=2)


            if  len(selection)==2:
                selected_columns = df.filter(like=selection[0])
                if not selected_columns.empty:


                    nombre_varindep = selection[0]
                    nombre_vardep = selection[1]


                    # Definir fechas de inicio y fin
                    start_date = dt.datetime(2022, 10, 1)
                    end_date = dt.datetime(2023, 3, 24)


                    # Crear slider de doble end
                    start_time, end_time = st.slider(
                        "Selecciona un rango de fechas",
                        value=(start_date, end_date),
                        format="YYYY/MM/DD - hh:mm",
                        step=dt.timedelta(hours=1),
                        min_value=start_date,
                        max_value=end_date)


                    st.write("<p style='color:#173763','font-family: Verdana, sans-
serif'>Rango de fechas seleccionado: {} a {}</p>".format(start_time, end_time),
unsafe_allow_html=True)


                    fecha=df['Fecha']
                    x = df[nombre_varindep]
                    y = df[nombre_vardep]
                    df = df.loc[(fecha<=end_time) & (fecha>=start_time),:]
                    x = df[nombre_varindep]
                    y = df[nombre_vardep]


                    if len(df.index)<10:
                        st.write(f"<p style='text-align: center; font-family: Verdana, sans-
serif; text-transform: uppercase; color: #173763;'><strong>AMPLÍE LA SELECCIÓN
TEMPORAL</strong></p>", unsafe_allow_html=True)
```

```
                else:

                #Poka Yoke: caso de los motores principales y sus velocidades de
ralentí, para eliminar ruido con datos erroneos u obtenidos durante la operación con
turbinas en buques CODOG

                    if nombre_vardep == 'DE1.speed' or nombre_vardep == 'DE2.speed':
                        ralenti=st.text_input('VELOCIDAD DE RALENTÍ (RPM)    ',value='0')
                        if ralenti.isdigit():
                            ralenti = int(ralenti)
                        else:
                            ralenti = None

                        x = df[nombre_varindep]
                        y = df[nombre_vardep]
                        df = df.loc[(y>=ralenti), :]
                        x = df[nombre_varindep]
                        y = df[nombre_vardep]


                    if nombre_varindep == 'DE1.speed' or nombre_varindep == 'DE2.speed':
                        ralenti = st.text_input('VELOCIDAD DE RALENTÍ (RPM)', value='0')
                        if ralenti.isdigit():
                            ralenti = int(ralenti)
                        else:
                            ralenti = None

                        x = df[nombre_varindep]
                        y = df[nombre_vardep]
                        df = df.loc[(x>=ralenti), :]
                        x = df[nombre_varindep]
                        y = df[nombre_vardep]

                #Poka Yoke: caso de las velocidades y sus velocidades operación con
turbina, para eliminar ruido con datos erroneos u obtenidos durante estas operaciones en
buques CODOG
```

```
                    if nombre_vardep == 'SHIP.overground.speed':
                        buque=st.radio('Tipo de buque',('F100','Lancha de
instrucción'),horizontal=True)
                        if buque == 'F100':
                            velocidad=st.text_input('VELOCIDAD DE OPERACIÓN CON TURBINA
(KTS):',value='40')
                            if velocidad.isdigit():
                                velocidad = int(velocidad)
                            else:
                                velocidad = None
                        else:
                            velocidad = 30

                        x = df[nombre_varindep]
                        y = df[nombre_vardep]
                        df = df.loc[(y<=velocidad), :]
                        x = df[nombre_varindep]
                        y = df[nombre_vardep]




                    if nombre_varindep == 'SHIP.overground.speed':
                        buque=st.radio('Tipo de buque',('F100','Lancha de
instrucción'),horizontal=True)
                        if buque == 'F100':
                            velocidad = st.text_input('VELOCIDAD DE OPERACIÓN CON
TURBINA (KTS): ', value='40')
                            if velocidad.isdigit():
                                velocidad = int(velocidad)
                            else:
                                velocidad = None
                        else:
                            velocidad = 30

                        x = df[nombre_varindep]
                        y = df[nombre_vardep]
                        df = df.loc[(x<=velocidad), :]
```

```
                        x = df[nombre_varindep]
                        y = df[nombre_vardep]

                    else:
                        x = df[nombre_varindep]
                        y = df[nombre_vardep]



                    xper=x
                    yper=y

                    if x.empty or y.empty:
                        st.write("Seleccione dos variables para compararlas")



                    # Modelado eligiendo el grado del polinomio que se ajustará a los
datos (a realizar por el usuario con el selector que aparece en pantalla)
                    grado = ['Grado 0', 'Grado 1', 'Grado 2', 'Grado 3', 'Grado 4',
'Grado 5', 'Grado 6','Grado 7']

                    # Análisis de datos
                    i = st.number_input('Grado del modelo',min_value=1,max_value=6,
step=1)

                    fig, ax = plt.subplots(figsize=(6, 4))



                    col1, col2= st.columns([2, 1])
                    with col1:
                        # Entrenamiento de prueba supervisado
                        X_train, X_test, y_train, y_test = train_test_split(x, y,
test_size=0.1)
                        X_train = X_train.values.reshape([X_train.values.shape[0], 1])
                        X_test = X_test.values.reshape([X_test.values.shape[0], 1])

                        # Gráfica de dispersión y ajuste de regresión
                        poly_features = PolynomialFeatures(degree=i, include_bias=False)
```

```
                            X_poly = poly_features.fit_transform(X_train)
                            Xt_poly = poly_features.fit_transform(X_test)

                            poly_model = LinearRegression()
                            poly_model.fit(X_poly, y_train)
                            pred = poly_model.predict(X_poly)
                            pred2 = poly_model.predict(Xt_poly)
                            xt, yt = zip(*sorted(zip(X_test, pred2)))
                            X, y = zip(*sorted(zip(X_train, pred)))

                            r2_train = r2_score(y_train, pred)
                            r2_test = r2_score(y_test, pred2)

                            clf = svm.OneClassSVM(nu=0.1, kernel="poly", degree=i)
                            clf.fit(X_train)
                            anomalias = clf.predict(X_train) == -1
                            x_anomalos = X_train[anomalias]
                            y_anomalos = y_train[anomalias]
                            x_normales = X_train[~anomalias]
                            y_normales = y_train[~anomalias]



                            ax.plot(X, y, '-', color='r', label=grado[i], markersize=2)
                            ax.plot(x_normales, y_normales, '+', color='darkblue',
markersize=5, label="datos normales", picker=True)
                            ax.plot(x_anomalos, y_anomalos, 'x', color='m', markersize=7,
label="datos anómalos", picker=True)



                            coefs = poly_model.coef_

                            # Expresamos el polinomio en la forma ax^n + bx^(n-1) + ... + zx
+ c
                            coefs_text = ''
                            for j in range(i-1, -2, -1):
                                if j == i-1:
```

```
                                coefs_text += f"{coefs[j]:.2f}x^{j+1}"
                            elif j == -1:
                                coefs_text += f" +
{round(poly_model.predict(np.array([[0] + [0]*(i-1)]))[0],2)}"
                            else:
                                coefs_text += f" + {coefs[j]:.2f}x^{j+1}"
                        ax.text(0.02, 0.95, coefs_text, transform=ax.transAxes,
fontsize=10, verticalalignment='top')


                        ax.legend(loc=2)
                        ax.text(0.02, 0.90, f"Número de anomalías: {len(x_anomalos)}",
transform=ax.transAxes, fontsize=10, verticalalignment='top')

                        # Personalización del gráfico
                        plt.legend()
                        plt.xlabel(nombre_varindep)
                        plt.ylabel(nombre_vardep)



                        st.pyplot(fig)
                        st.text(f"R2 Train: {r2_train:.2f}\nR2 Test:
{r2_test:.2f}\nPolinomio: {coefs_text}")



                    with col2:
                        filtro1=st.checkbox(label="FILTRO EN PRIMER CUARTIL")
                        filtro2=st.checkbox(label="FILTRO EN SEGUNDO CUARTIL")
                        filtro3=st.checkbox(label="FILTRO EN TERCER CUARTIL")
                        filtro4=st.checkbox(label="FILTRO EN CUARTO CUARTIL")


                        Qx1=xper.quantile(0.25)
                        Qx2=xper.quantile(0.5)
                        Qx3=xper.quantile(0.75)
                        Qx4=xper.quantile(1)


                        filtro=st.slider(label="Permisividad", min_value=0.0,
max_value=1.5, value=1.5)
```

```
                        if filtro1:
                            Qx1_valores = df.loc[xper < Qx1, nombre_varindep]

                            y_x1 = df.loc[xper < Qx1, nombre_vardep]

                            #Selección de los datos considerados como ruido
                            Qy_x11=y_x1.quantile(0.25)
                            Qy_x13=y_x1.quantile(0.75)

                            #Rango intercuartílico
                            IQR=Qy_x13-Qy_x11

                            #Cálculo de los bigotes inferior y superior del box-plot
                            BI1=Qy_x11 - filtro*IQR
                            BS1=Qy_x13 + filtro*IQR

                            nuevo_dfx1 = df.loc[(BI1 <= yper) & (yper <=
BS1),nombre_varindep]
                            nuevo_dfy1 = df.loc[(BI1 <= yper) & (yper <=
BS1),nombre_vardep]

                        else:
                            nuevo_dfx1 =  df.loc[xper <= Qx1, nombre_varindep]
                            nuevo_dfy1 = df.loc[xper <= Qx1, nombre_vardep]

                        if filtro2:
                            Qx2_valores = df.loc[(Qx1 < xper) & (xper <= Qx2),
nombre_varindep]

                            y_x2 = df.loc[(Qx1 < xper) & (xper <= Qx2), nombre_vardep]

                            #Selección de los datos considerados como ruido
                            Qy_x21=y_x2.quantile(0.25)
                            Qy_x23=y_x2.quantile(0.75)

                            #Rango intercuartílico
                            IQR=Qy_x23-Qy_x21
```

```
                            #Cálculo de los bigotes inferior y superior del box-plot
                            BI2=Qy_x21 - filtro*IQR
                            BS2=Qy_x23 + filtro*IQR

                            nuevo_dfx2 = df.loc[(BI2 <= yper) & (yper <=
BS2),nombre_varindep]
                            nuevo_dfy2 = df.loc[(BI2 <= yper) & (yper <=
BS2),nombre_vardep]

                        else:
                            nuevo_dfx2 =df.loc[(Qx1 < xper) & (xper <= Qx2),
nombre_varindep]
                            nuevo_dfy2 = df.loc[(Qx1 < xper) & (xper <= Qx2),
nombre_vardep]

                        if filtro3:
                            Qx3_valores = df.loc[(Qx2 < xper) & (xper <= Qx3),
nombre_varindep]

                            y_x3 = df.loc[(Qx2 < xper) & (xper <= Qx3), nombre_vardep]

                            #Selección de los datos considerados como ruido
                            Qy_x31=y_x3.quantile(0.25)
                            Qy_x33=y_x3.quantile(0.75)

                            #Rango intercuartílico
                            IQR=Qy_x33-Qy_x31

                            #Cálculo de los bigotes inferior y superior del box-plot
                            BI3=Qy_x31 - filtro*IQR
                            BS3=Qy_x33 + filtro*IQR

                            nuevo_dfx3 = df.loc[(BI3 <= yper) & (yper <=
BS3),nombre_varindep]
                            nuevo_dfy3 = df.loc[(BI3 <= yper) & (yper <=
BS3),nombre_vardep]

                        else:
```

```
                                nuevo_dfx3 =df.loc[(Qx2 < xper) & (xper <= Qx3),
nombre_varindep]
                                nuevo_dfy3 =df.loc[(Qx2 < xper) & (xper <= Qx3),
nombre_vardep]

                            if filtro4:
                                Qx4_valores =df.loc[(Qx3 < xper) & (xper <= Qx4),
nombre_varindep]

                                y_x4 = df.loc[(Qx3 < xper) & (xper <= Qx4), nombre_vardep]

                                #Selección de los datos considerados como ruido
                                Qy_x41=y_x4.quantile(0.25)
                                Qy_x43=y_x4.quantile(0.75)

                                #Rango intercuartílico
                                IQR=Qy_x43-Qy_x41

                                #Cálculo de los bigotes inferior y superior del box-plot
                                BI4=Qy_x41 - filtro*IQR
                                BS4=Qy_x43 + filtro*IQR

                                nuevo_dfx4 = df.loc[(BI4 <= yper) & (yper <=
BS4),nombre_varindep]
                                nuevo_dfy4 = df.loc[(BI4 <= yper) & (yper <=
BS4),nombre_vardep]

                            else:
                                nuevo_dfx4 = df.loc[(Qx3 < xper) & (xper <= Qx4),
nombre_varindep]
                                nuevo_dfy4 = df.loc[(Qx3 < xper) & (xper <= Qx4),
nombre_vardep]



nuevofiltrox=pd.concat([nuevo_dfx1,nuevo_dfx2,nuevo_dfx3,nuevo_dfx4],axis=0)

nuevofiltroy=pd.concat([nuevo_dfy1,nuevo_dfy2,nuevo_dfy3,nuevo_dfy4],axis=0)
```

```
                            # Entrenamiento de prueba supervisado
                            X_train, X_test, y_train, y_test =
train_test_split(nuevofiltrox, nuevofiltroy, test_size=0.1)
                            X_train = X_train.values.reshape([X_train.values.shape[0], 1])
                            X_test = X_test.values.reshape([X_test.values.shape[0], 1])

                            # Gráfica de dispersión y ajuste de regresión
                            poly_features = PolynomialFeatures(degree=i, include_bias=False)

                            X_poly = poly_features.fit_transform(X_train)
                            Xt_poly = poly_features.fit_transform(X_test)

                            poly_model = LinearRegression()
                            poly_model.fit(X_poly, y_train)
                            pred = poly_model.predict(X_poly)
                            pred2 = poly_model.predict(Xt_poly)
                            xt2, yt2 = zip(*sorted(zip(X_test, pred2)))
                            X2, y2 = zip(*sorted(zip(X_train, pred)))

                            r2_train = r2_score(y_train, pred)
                            r2_test = r2_score(y_test, pred2)



                            #Caracterización de datos anómalos con respecto al modelo
seleccionado
                            residuos = y_train - pred

                            # Definimos un umbral para detectar anomalías
                            umbral = np.percentile(residuos, 95)

                            # Filtramos los datos anómalos
                            x_anomalos = X_train[residuos > umbral]
                            y_anomalos = y_train[residuos > umbral]
                            x_normales = X_train[residuos <= umbral]
```

```
                            y_normales = y_train[residuos <= umbral]


                            fig2, ax2 = plt.subplots(figsize=(6, 4))
                            ax2.plot(X, y, '-', color='r', label=grado[i], markersize=2)
                            ax2.plot(x_normales, y_normales, '+', color='darkblue',
markersize=5, label="datos normales")
                            ax2.plot(x_anomalos, y_anomalos, 'x', color='m', markersize=7,
label="datos anómalos")









                            coefs = poly_model.coef_



                            # Expresamos el polinomio en la forma ax^n + bx^(n-1) + ... + zx
+ c
                            coefs_text = ''
                            for j in range(i-1, -2, -1):
                                if j == i-1:
                                    coefs_text += f"{coefs[j]:.8f}x^{j+1}"
                                elif j == -1:
                                    coefs_text += f" +
{round(poly_model.predict(np.array([[0] + [0]*(i-1)]))[0],2)}"
                                else:
                                    coefs_text += f" + {coefs[j]:.6f}x^{j+1}"
                            ax.text(0.02, 0.95, coefs_text, transform=ax.transAxes,
fontsize=10, verticalalignment='top')



                            ax2.legend(loc=2)



                            # Personalización del gráfico
                            plt.legend()
                            plt.xlabel(nombre_varindep)
```

```
                            plt.ylabel(nombre_vardep)
                            st.pyplot(fig2)

                            st.text(f"R2 Train: {r2_train:.2f}\nR2 Test:
{r2_test:.2f}\nPolinomio: {coefs_text}")

                            guardarmodelo = st.button('GUARDAR MODELO',
use_container_width=True)
                            if guardarmodelo:

                                data={f'{nombre_varindep}':X2,f'{nombre_vardep}':y2}
                                modelo = pd.DataFrame(data)
                                with pd.ExcelWriter(excel_file,
mode='a',if_sheet_exists='replace') as writer:
                                    modelo.to_excel(writer, sheet_name=f'{nombre_vardep}''-
'f'{nombre_varindep}')



                        # Definimos el slider y el zoom
                        ratio = st.slider(label="", min_value=1, max_value=99, value=50,
format='')
                        zoom = 50 / ratio

                        col1, col2 = st.columns([1, zoom])
                        with col1:
                            fig1, ax1 = plt.subplots(figsize=(6, 5))
                            fig1=px.histogram(yper, nbins=20, labels={'Frecuencia':'Conteo',
'x':'Valores'})
                            st.plotly_chart(fig1)


                        with col2:
                            fig2, ax2 = plt.subplots(figsize=(4, 5))
                            fig2 = px.box(yper)
                            st.plotly_chart(fig2)
```

```
                        st.markdown("<h1 style='text-align: center;'>VISOR DE MODELOS</h1>",
unsafe_allow_html=True)


                        col1, col2 = st.columns([2, 1])


                        with col2:
                            st.markdown("<h3 style='text-align: center;'>COMPROBADOR DE
ANOMALÍAS</h3>", unsafe_allow_html=True)



                            st.markdown(f"<p style='vertical-align:
middle;'>{nombre_varindep}:</p>", unsafe_allow_html=True)
                            compx = st.text_input("x",label_visibility="hidden")
                            if compx.isdigit():
                                compx = int(compx)
                            else:
                                compx = None


                            st.markdown(f"<p style='vertical-align:
middle;'>{nombre_vardep}:</p>", unsafe_allow_html=True)
                            compy = st.text_input("y",label_visibility="hidden")
                            if compy.isdigit():
                                compy = int(compy)
                            else:
                                compy = None


                            st.markdown("<h3 style='text-align: center;'> PREDICCIÓN</h3>",
unsafe_allow_html=True)
                            st.markdown(f"<p style='vertical-align:
middle;'>{nombre_varindep}:</p>", unsafe_allow_html=True)
                            predin = st.text_input("xpred",label_visibility="hidden")



                            if predin.strip() and predin.replace('.', '', 1).isdigit():
                                predin = float(predin)


                                predin=pd.DataFrame([predin],columns=['predin'])
                                predin2d = predin.values.reshape([predin.values.shape[0],
1])
```

```
                                predin_poly = poly_features.fit_transform(predin2d)
                                pp=1

                                predout=poly_model.predict(predin_poly)

                                st.text(f"Predicción para {nombre_vardep}:\n ")
                                st.write(f"<p style='text-align: center; font-family:
Verdana, sans-serif; text-transform: uppercase; color: #173763;'><strong> {predout[-
1]:.2f}</strong></p>",
                                unsafe_allow_html=True)

                            else:
                                predin = None
                                pp=0
                        with col1:
                            fig, ax = plt.subplots(figsize=(6, 4))

                            ax.plot(X2, y2, '-', color='darkblue', label="Modelo",
markersize=2)

                            sns.regplot(x=nuevofiltrox,y=nuevofiltroy,
ci=95,scatter=False,order=i,line_kws={"alpha":0},label='Distribución de datos')

                            if compx and compy:
                                ax.plot(compx, compy, '.', color='r', label="Comprobación",
markersize=10)
                                ax.vlines(compx,ymin=0,ymax=max(y2), color='r')
                                ax.hlines(compy,xmin=0,xmax=max(X2), color='r')

                                bbox = dict(boxstyle="round", fc="0.8")
                                arrowprops = dict(arrowstyle="-
>",connectionstyle="angle,angleA=0,angleB=90,rad=10")
                                offset = 72

                                predcomp = float(compx)

                                predcomp=pd.DataFrame([predcomp],columns=['predcomp'])
                                predcomp2d =
predcomp.values.reshape([predcomp.values.shape[0], 1])
```

```
                                predcomp_poly = poly_features.fit_transform(predcomp2d)



                                predcompout=poly_model.predict(predcomp_poly)


                                ax.annotate(
                                    f'$\Delta$y{float(predcompout[-1])-compy:.2f}',
                                    (compx, compy),
                                    xytext=(-1*offset, offset), textcoords='offset points',
                                    bbox=bbox, arrowprops=arrowprops)



                            if pp==1:

                                ax.plot(predin, predout, '.', color='b',
label="Comprobación", markersize=15, alpha=0.5)

                            ax.legend(loc="best")
                            ax.set_xlabel(nombre_varindep)
                            ax.set_ylabel(nombre_vardep)
                            ax.set_xlim([min(X2), max(X2)])
                            ax.set_ylim([min(y2), max(y2)])
                            st.pyplot(fig)




                else:
                    st.write(f"<p style='text-align: center; font-family: Verdana, sans-serif;
text-transform: uppercase; color: #173763;'><strong>SELECCIONE DOS VARIABLES PARA
COMPARARLAS</strong></p>",
                    unsafe_allow_html=True)


                    import graphviz
                    graph = graphviz.Digraph()
                    graph.attr('node', shape='rectangle', style='filled', color='#173763',
fontcolor='white')
                    graph.attr('edge', color='darkblue')
```

```
graph.attr(rankdir='LR', nodesep='0.5', ranksep='0.5')
graph.edge('SELECCIÓN DE ARCHIVO', 'SELECCIÓN DE RANGO TEMPORAL')
graph.edge('SELECCIÓN DE ARCHIVO', 'SELECCIÓN DE VARIABLES')
graph.edge('SELECCIÓN DE RANGO TEMPORAL', 'FILTRADO')
graph.edge('SELECCIÓN DE VARIABLES', 'FILTRADO')
graph.edge('FILTRADO', 'MODELO')
st.graphviz_chart(graph)
```

ANEXO VIII: CÓDIGO FUENTE PÁGINA VISUALIZACIÓN

```
import streamlit as st
from PIL import Image
import subprocess


# Configuración inicial de la página
st.set_page_config(page_title="Programa de seguimiento de las LLII de la ENM",
page_icon='.', layout="wide", initial_sidebar_state="auto")

# Selección del tema, con los colores marcados por el Manual de Imagen Corporativa del
AJEMA
theme = """
[theme]
base="light"
primaryColor="#173763"
secondaryBackgroundColor="#c7c7c7"

"""

st.write(f'<style>{theme}</style>', unsafe_allow_html=True)


# Llamamiento a las imáges de los logos de la ENM y Armada para incluir en la barra lateral
image = Image.open(' /IMAGES/escudo-enm.png')
st.sidebar.image(image, width=150)
image = Image.open(' /IMAGES/IMAGOTIPOARMADA.png')
st.sidebar.image(image, width=150)

#Título y subtítulo de la página
st.header('Ventana de visualización de datos')
st.subheader('AI de la serie Rodman 66')


# Dirección al repositorio donde se encuentra el ejecutable para la visualización en 3D

ruta_ejecutable = "INTRODUCIR RUTA EJECUTABLE"
if st.button("Iniciar Visualización inmersiva"):
    subprocess.run(["open", "-a", "ruta_ejecutable"])
```

ANEXO IX: CÓDIGO FUENTE PÁGINA ABOUT

```
import streamlit as st
from PIL import Image

# Configuración inicial de la página
st.set_page_config(page_title="Programa de seguimiento de las LLII de la ENM",
page_icon=None, layout="wide", initial_sidebar_state="auto")

# Selección del tema, con los colores marcados por el Manual de Imagen Corporativa del
AJEMA
theme = """
[theme]
base="light"
primaryColor="#173763"
secondaryBackgroundColor="#c7c7c7"

"""

st.write(f'<style>{theme}</style>', unsafe_allow_html=True)



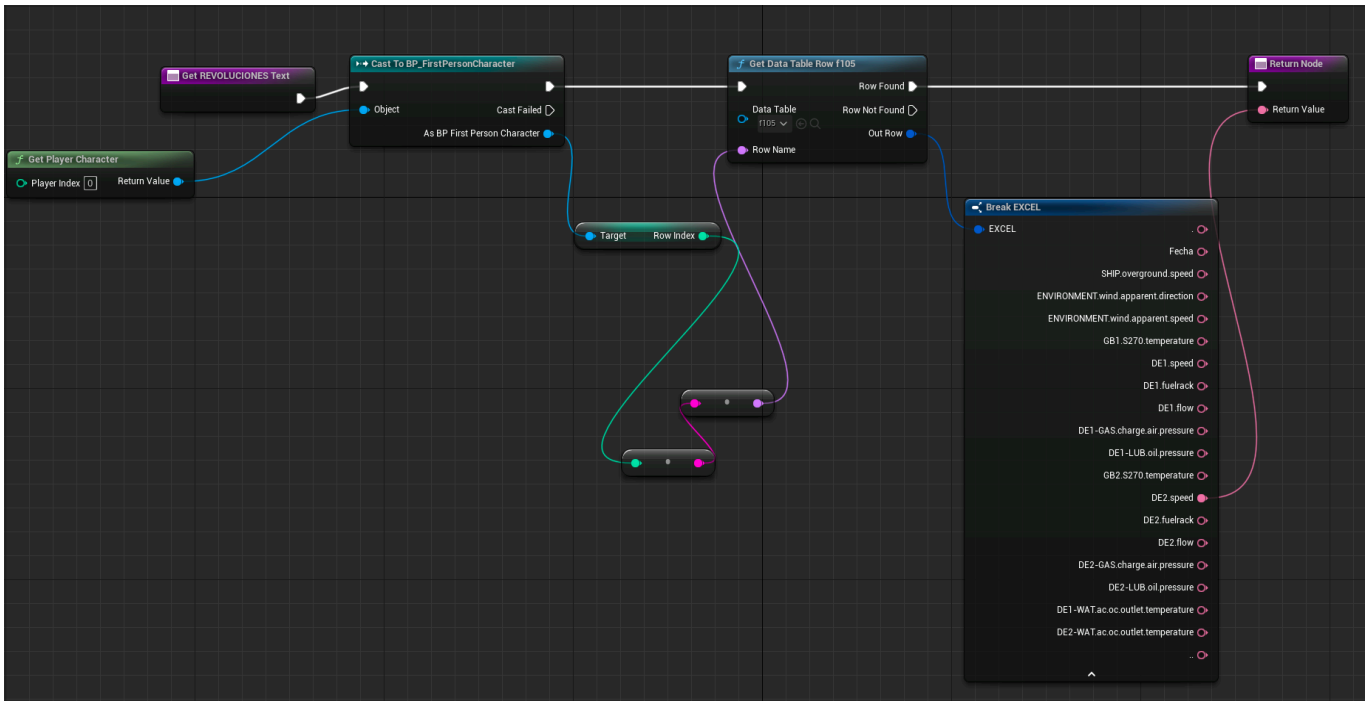
# Llamamiento a las imáges de los logos de la ENM y Armada para incluir en la barra lateral
image = Image.open(' /IMAGES/escudo-enm.png')
st.sidebar.image(image, width=150)
image = Image.open(' /IMAGES/IMAGOTIPOARMADA.png')
st.sidebar.image(image, width=150)

#Título y subtítulo
st.header('Sobre la aplicación')
st.subheader('AF (CGA-EOF) Francisco Antón Antón')

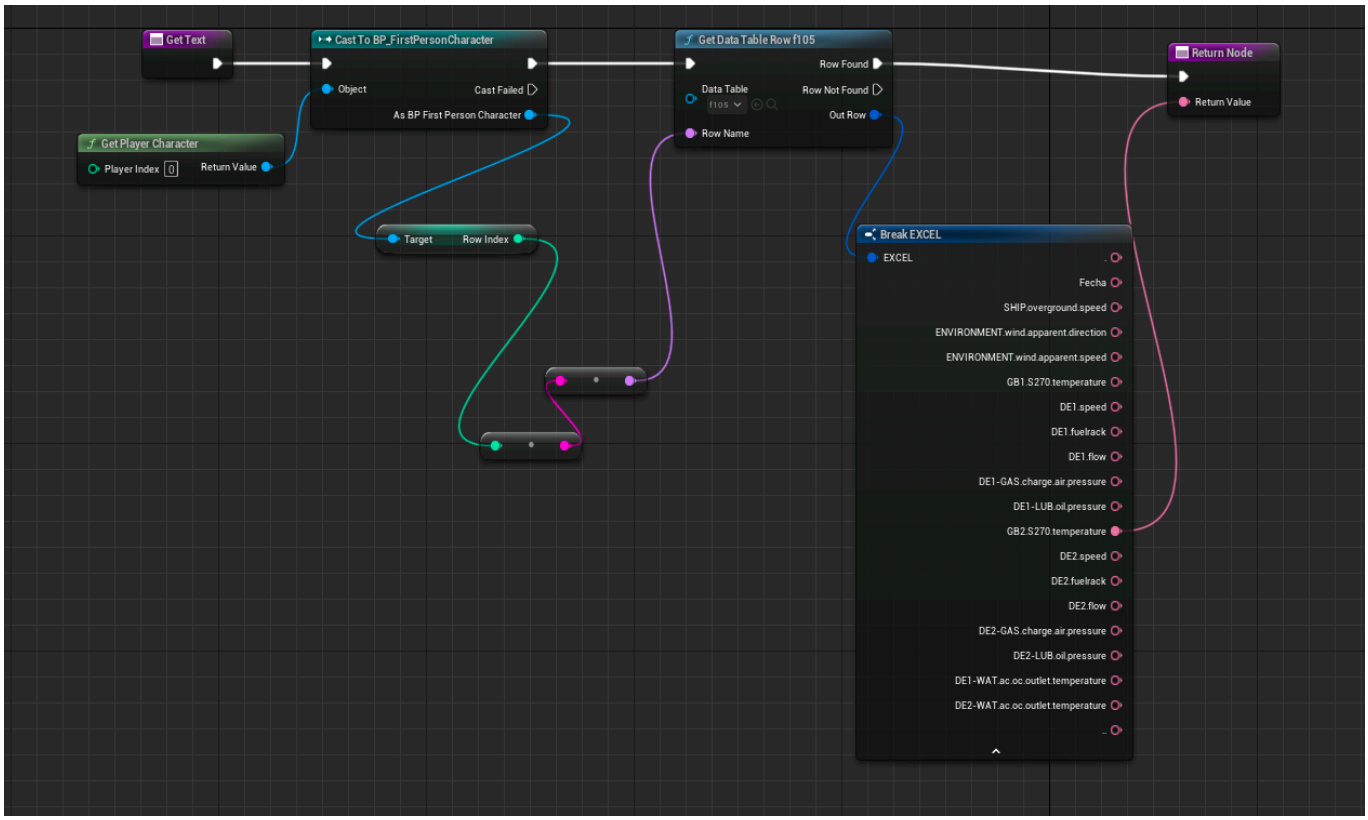
#Texto a incluir
st.write("TEXTO INTRODUCCIÓN")
```

ANEXO X: BLUEPRINTS UE5 WIDGET MARINE POWER DISPLAY [ER (1) Y BR(2)]

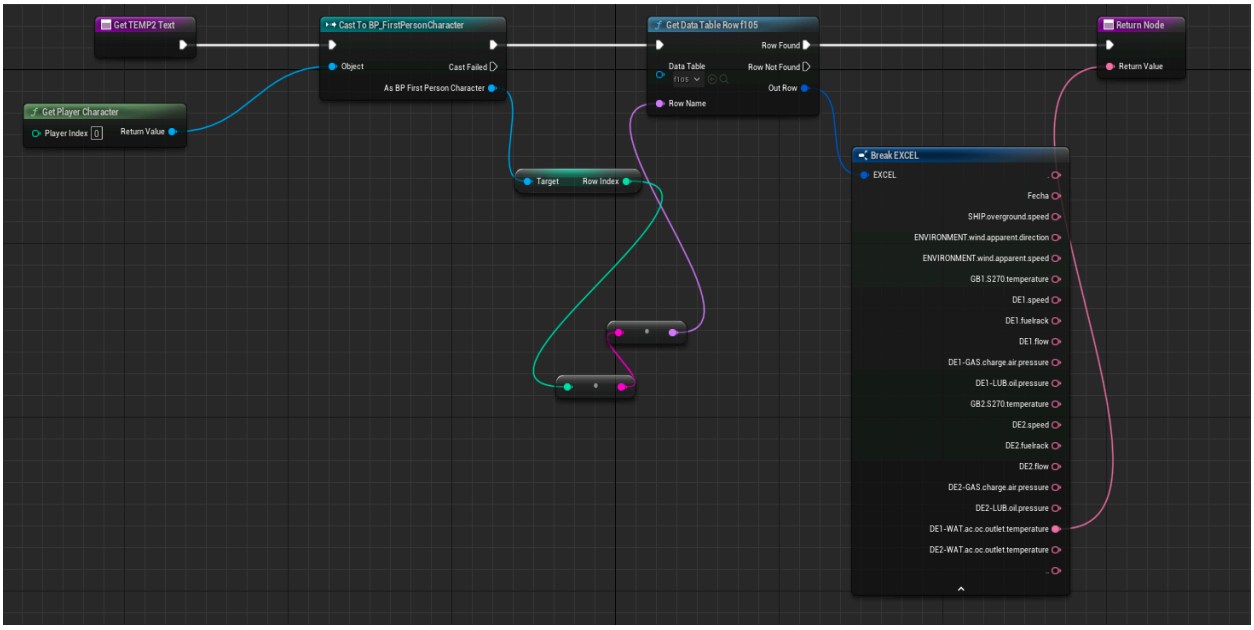
REVOLUCIONES



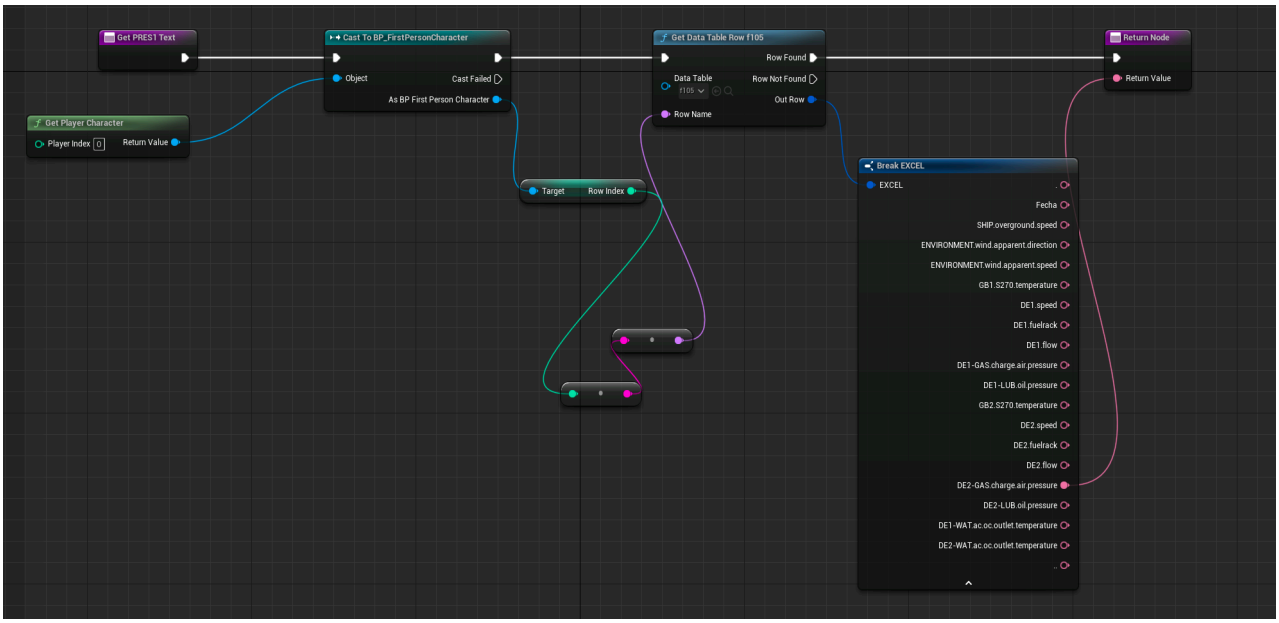
TEMPERATURA ACEITE DE LUBRICACIÓN



TEMPERATURA AGUA DE REFRIGERACIÓN

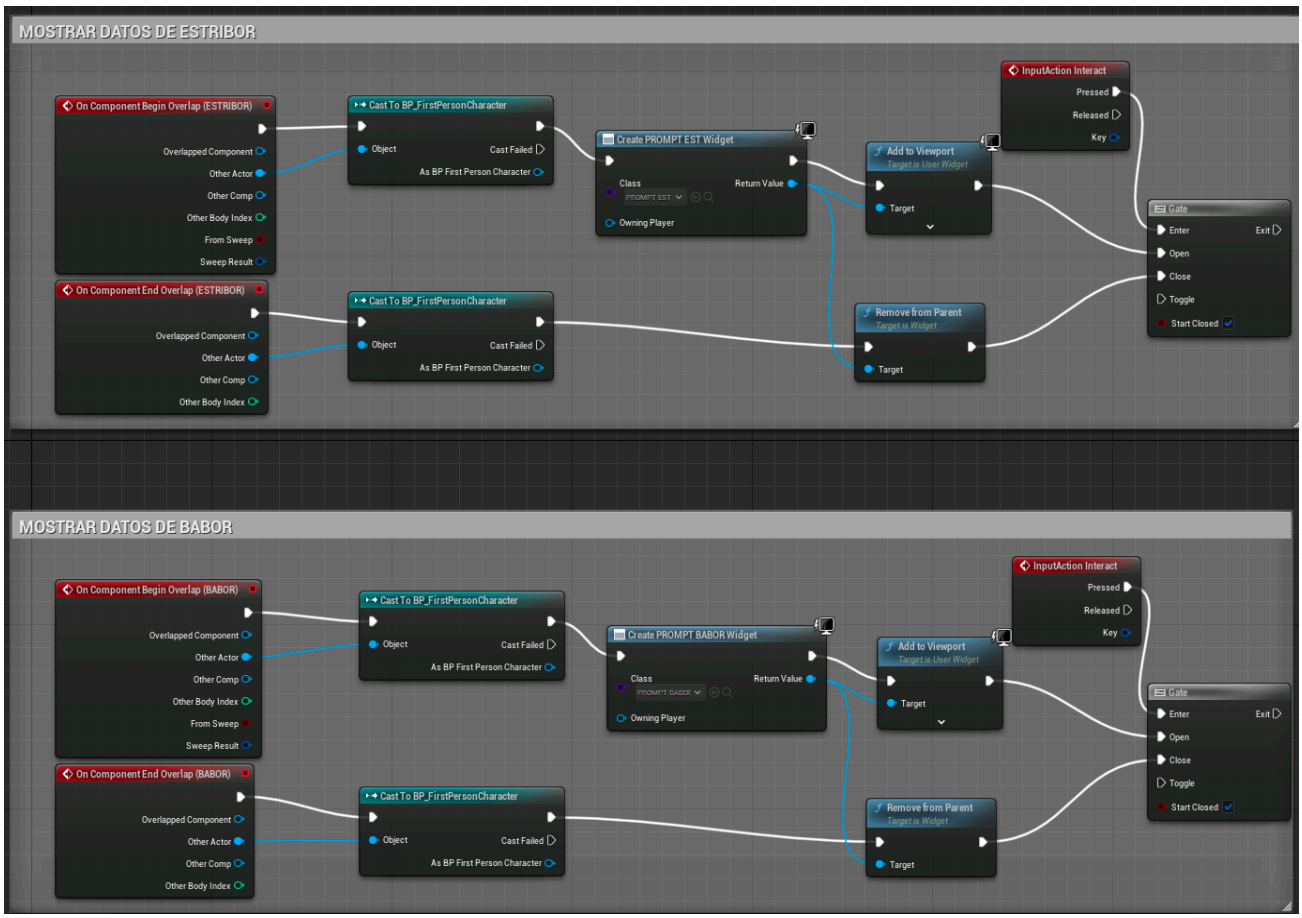


PRESIÓN 1

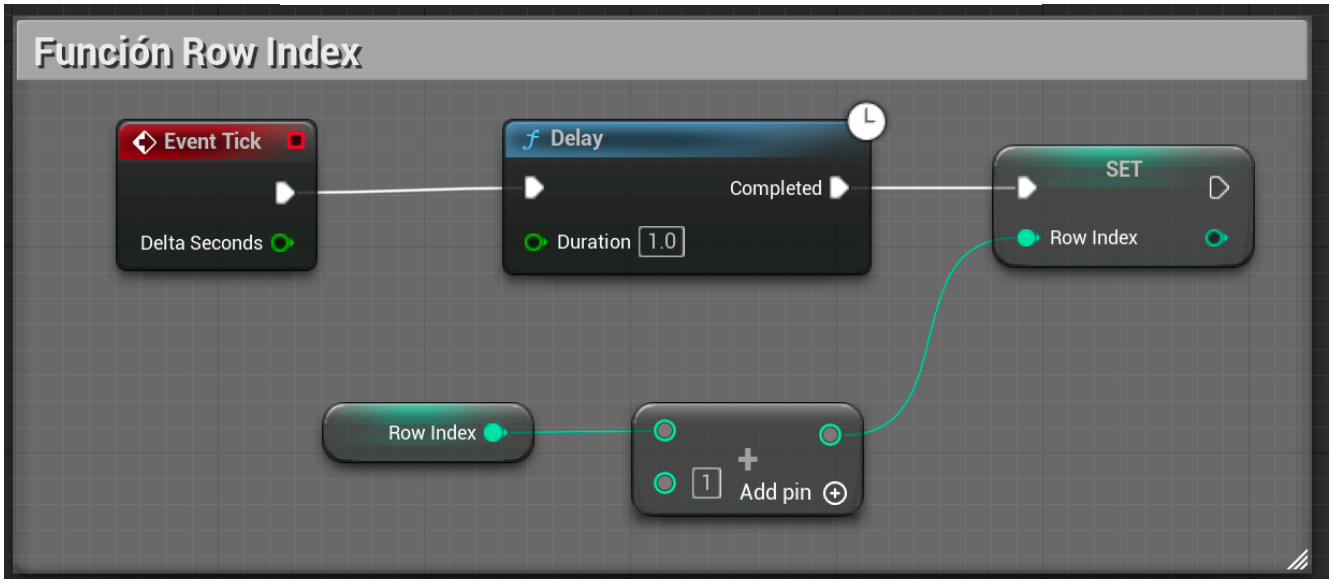


ANEXO XI: BLUEPRINTS UE5 INTERACCIÓN

Desde Blueprint de la malla estática de los equipos del puente



Desde Blueprint del controlador en primera persona



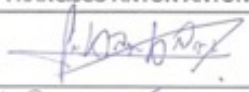
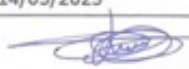
ANEXO XII: DOCUMENTACIÓN OPERATIVA PARA LA TOMA DE DATOS A BORDO DE LAS LLII

El presente documento es un escaneado del original en poder del autor.



	<< PRUEBA PROGRESIVA DE VELOCIDAD >>			
	Prueba Nº: 1	Clase: R-66	Serie: 1	Numeral: A-121
	Hecho: AF-FAA	Revisado: CF-GGC	V°. B°.: TN-PCP	H 1 de 6 HH
	Fecha:05/03/23	Fecha: 08/03/23	Fecha: 10/03/23	Rev.: A

CONTROL DE REVISIONES		
REVISIÓN	DESCRIPCIÓN	HOJAS
A	EMISIÓN FINAL DE LA PRUEBA	TODAS

CERTIFICACIÓN DE EJECUCIÓN DE PRUEBA	
CENTRO UNIVERSITARIO DE LA DEFENSA EN LA ESCUELA NAVAL MILITAR EN MARÍN	
BUQUE AL QUE APLICA	LANCHA DE INSTRUCCIÓN A-121
REALIZADA POR:	AF FRANCISCO ANTÓN ANTÓN
FIRMA	
SUPERVISADA POR:	AN DOMINGUEZ Gómez (El Comandante)
FECHA	14/03/2023
FIRMA	
V°B° DIRECTOR TFG:	TN PEDRO CARRASCO PEÑA
FECHA	15/03/2023
FIRMA	

CUD-ENM	TRABAJO FIN DE GRADO CURSO 22/23	PROMOCIÓN:
Título del TFG:	Monitorización de variables de influencia en operación de buques para su modelado y visualización.	
Autor:	AF Francisco Antón Antón.	

	<< PRUEBA PROGRESIVA DE VELOCIDAD >>			
	Prueba Nº: 1	Clase: R-66	Serie: 1	Numeral: A-121
	Hecho: AF-FAA	Revisado: CF-GGC	V°. B°.: TN-PCP	H 2 de 6 HH
	Fecha:05/03/23	Fecha: 08/03/23	Fecha: 10/03/23	Rev.: A

1 HOJA DE OBSERVACIONES

- Viento del SSW - 13kts (rachas 18 kts)
- Prueba de Máquinas y Arranque de Proas S/N
- Los datos recogidos se encuentran Anexos al protocolo

CUD-ENM	TRABAJO FIN DE GRADO CURSO 22/23	PROMOCIÓN:
Título del TFG:	Monitorización de variables de influencia en operación de buques para su modelado y visualización.	
Autor:	AF Francisco Antón Antón.	

<< PRUEBA PROGRESIVA DE VELOCIDAD >>			
Prueba Nº: 1	Clase: R-66	Serie: 1	Numeral: A-121
Hecho: AF-FAA	Revisado: CF-GGC	V°. B°.: TN-PCP	H 3 de 6 HH
Fecha:05/03/23	Fecha: 08/03/23	Fecha: 10/03/23	Rev.: A

2 ÍNDICE

1 HOJA DE COMENTARIOS: 2
2 ÍNDICE 3
3 DOCUMENTACIÓN DE REFERENCIA 4
4 PRUEBAS PREVIAS 4
5 COMPONENTES Y CONFIGURACIÓN 4
6 OBJETO 4
7 PROCEDIMIENTO 4
7.1 PRECAUCIONES DE SEGURIDAD 4
7.2 CONDICIONES DE EJECUCIÓN DE LA PRUEBA 5
7.3 DESCRIPCIÓN DEL PROCEDIMIENTO 5
8 RECURSOS MATERIALES PARA LA PRUEBA 6
9 PROCEDIMIENTO "A" 6

CUD-ENM	TRABAJO FIN DE GRADO CURSO 22/23	PROMOCIÓN:
Título del TFG:	Monitorización de variables de influencia en operación de buques para su modelado y visualización.	
Autor:	AF Francisco Antón Antón.	

	<< PRUEBA PROGRESIVA DE VELOCIDAD >>			
	Prueba Nº: 1	Clase: R-66	Serie: 1	Numeral: A-121
	Hecho: AF-FAA	Revisado: CF-GGC	Vº. Bº.: TN-PCP	H 4 de 6 HH
	Fecha:05/03/23	Fecha: 08/03/23	Fecha: 10/03/23	Rev.: A

3 DOCUMENTACIÓN DE REFERENCIA

La documentación de referencia para la presente prueba es la que a continuación se relaciona:

(1) Publicación 4810 Escuela Naval Militar, Manual de Lanchas de Instrucción.
(2) Protocolo de pruebas R-66 Astillero.
(3) CATERPILLAR Marine Power Display User guide.

4 PRUEBAS PREVIAS

Se comprobará el correcto funcionamiento de los siguientes elementos como elemento fundamental para efectuar las pruebas de este protocolo:

(1) Arrancado de motores.
(2) Prueba sobre Amarras.

5 COMPONENTES Y CONFIGURACIÓN

Se comprobará el correcto funcionamiento de los siguientes componentes y configuración:

(1) CAT Marine Power Display ER (MPD1)/ Presentación modo normal
(2) CAT Marine Power Display BR (MPD2)/Presentación modo normal
(3) Sistemas de control de propulsión/ Modo SYNC. en régimen de revoluciones.

6 OBJETO

El objeto de la presente prueba es la medición de las variables de operación del sistema lancha a bordo de la plataforma en distintos regímenes de funcionamiento considerando las condiciones de viento y mar.

7 PROCEDIMIENTO

7.1 PRECAUCIONES DE SEGURIDAD

El personal de dotación franco permanecerá en interiores durante las progresiones de velocidad.

El personal que tenga que permanecer en cubierta, portarán chalecos salvavidas en todo momento y línea de vida si es necesario.

CUD-ENM	TRABAJO FIN DE GRADO CURSO 22/23	PROMOCIÓN:
Título del TFG:	Monitorización de variables de influencia en operación de buques para su modelado y visualización.	
Autor:	AF Francisco Antón Antón.	

	<< PRUEBA PROGRESIVA DE VELOCIDAD >>			
	Prueba Nº: 1	Clase: R-66	Serie: 1	Numeral: A-121
	Hecho: AF-FAA	Revisado: CF-GGC	V°. B°.: TN-PCP	H 5 de 6 HH
	Fecha:05/03/23	Fecha: 08/03/23	Fecha: 10/03/23	Rev.: A

7.2 CONDICIONES DE EJECUCIÓN DE LA PRUEBA

Durante la prueba, el viento y el mar deberán estar lo más en calma posible. Los resultados se considerarán aceptables si la velocidad absoluta del viento es menor de 20 nudos y la corriente no es superior a 1 m/s. La profundidad de las aguas no será inferior de los 10 m para evitar el efecto de fondo.

7.3 DESCRIPCIÓN DEL PROCEDIMIENTO

PASO	DESCRIPCIÓN
	CONDICIONES INICIALES Y ENCENDIDO
1.	La prueba se realizará en la condición de desplazamiento en lastre y a los regímenes de funcionamiento siguientes: - Regímenes de revoluciones de 600, 700, 800, 9000, 1000, 1200 y 1500 rpm. NOTA: Caso de no poder hacerse en condición de lastre se anotará el desplazamiento de prueba.

PASO	DESCRIPCIÓN	RESULTADOS	
	PROCEDIMIENTO	Satisfactorio	No satisfactorio
2.	Registro de datos de ambos MPD en vídeo durante las evoluciones.	✓	
3.	La maniobra aceleración no deberá maniobrar con ángulos de timón superiores a 5° de caña.	✓	
4.	El rumbo del tramo de aceleración no presentará variaciones de más de 10° con respecto al rumbo inicial.	✓	
5.	Durante las distintas corridas se tomarán los datos concernientes a las condiciones ambientales y a la determinación de velocidad y condiciones de funcionamiento de la planta propulsora.	✓	
6.	Los tiempos de ejecución deben ser los establecidos en el procedimiento "A" (punto 9)	✓	

CUD-ENM	TRABAJO FIN DE GRADO CURSO 22/23	PROMOCIÓN:
Título del TFG:	Monitorización de variables de influencia en operación de buques para su modelado y visualización.	
Autor:	AF Francisco Antón Antón.	

	<< PRUEBA PROGRESIVA DE VELOCIDAD >>			
	Prueba Nº: 1	Clase: R-66	Serie: 1	Numeral: A-121
	Hecho: AF-FAA	Revisado: CF-GGC	V°. B°.: TN-PCP	H 6 de 6 HH
	Fecha:05/03/23	Fecha: 08/03/23	Fecha: 10/03/23	Rev.: A

8 RECURSOS MATERIALES PARA LA PRUEBA

SE UTILIZAN LOS INDICADORES INCORPORADOS AL EQUIPO/ SISTEMA

APARATO	ENGINE DISPLAY
MODELO	MARINE POWER DISPLAY
FABRICANTE	CAT
FECHA INCORPORACIÓN	2006
VALIDEZ	N/A

9 PROCEDIMIENTO "A"

ORDEN	TIEMPO
AV. 600 RPM BR/ER	1 min
AV. 700 RPM BR/ER	1 min
AV. 800 RPM BR/ER	1 min
AV. 900 RPM BR/ER	1 min
AV. 1000 RPM BR/ER	1 min
AV. 1200 RPM BR/ER	1 min
AV. 1500 RPM BR/ER	1 min
REPOSICIONAMIENTO PARA SEGUNDA PASADA	

NOTA: La toma de datos se realizará mediante grabación de los datos ofrecidos por los sistemas. Véase hoja de comentarios y anexos de la prueba.

CUD-ENM	TRABAJO FIN DE GRADO CURSO 22/23	PROMOCIÓN:
Título del TFG:	Monitorización de variables de influencia en operación de buques para su modelado y visualización.	
Autor:	AF Francisco Antón Antón.	

Fecha	SHIP.overground.speed	DE1.speed	DE1.fuelrack	DE1.flow	DE1-GAS.charge.air.pressure	DE1-LUB.oil.pressure	DE1-WAT.ac.oc.outlet.temperature	DE2-WAT.ac.oc.outlet.temperature	DE2.speed	DE2.fuelrack	DE2.flow	DE2-GAS.charge.air.pressure	DE2-LUB.oil.pressure	DE1-WAT.ac.oc.outlet.temperature	DE2-WAT.ac.oc.outlet.temperature
14/3/23 17:00	6	600	16	10	2320	240	71	34	601	15	12	2304	218	78	32
14/3/23 17:00	6,5	600	16	10	2320	244	72	34	600	15	14	2288	208	78	32
14/3/23 17:00	6,7	599	17	12	2320	240	72	34	599	15	12	2288	208	78	32
14/3/23 17:00	6	600	17	11	2320	240	72	34	600	15	9	2288	212	78	32
14/3/23 17:01	7	599	16	11	2320	236	72	34	599	15	10	2288	216	79	32
14/3/23 17:01	7	699	19	12	2336	304	72	34	700	16	16	2304	252	79	32
14/3/23 17:01	6,5	701	20	12	2336	296	73	34	699	20	16	2304	272	79	32
14/3/23 17:01	7	699	22	11	2336	288	73	34	699	20	12	2304	264	79	32
14/3/23 17:02	6,8	774	26	14	2336	340	73	34	698	17	18	2304	252	79	32
14/3/23 17:02	7,3	799	26	18	2336	340	74	34	800	26	20	2320	318	79	32
14/3/23 17:02	7,2	800	26	17	2336	348	74	34	798	26	18	2320	320	80	32
14/3/23 17:02	7,6	800	27	18	2336	348	75	34	798	26	18	2320	308	80	32
14/3/23 17:03	7,6	800	28	18	2340	340	75	34	799	26	18	2320	302	80	32
14/3/23 17:03	8,3	899	29	20	2352	404	76	34	898	29	18	2320	364	80	32
14/3/23 17:03	8,5	901	29	21	2352	408	76	34	900	29	18	2320	356	80	32
14/3/23 17:03	8,2	900	29	20	2352	408	77	34	898	29	18	2320	356	80	32
14/3/23 17:04	8,5	899	30	20	2352	382	77	34	897	29	18	2320	352	80	32
14/3/23 17:04	9	1000	36	22	2352	464	77	34	1000	33	20	2320	440	80	32
14/3/23 17:04	9,1	999	38	22	2352	450	76	35	997	33	20	2320	412	80	32
14/3/23 17:04	9,1	999	39	24	2352	452	78	35	999	32	20	2336	398	80	32
14/3/23 17:05	9,2	1000	39	25	2352	412	78	34	1001	32	22	2320	412	80	32
14/3/23 17:05	9,5	1200	43	35	2368	490	79	35	1199	36	24	2336	476	81	32
14/3/23 17:05	9,6	1200	45	34	2368	496	79	35	1198	36	24	2336	484	81	32
14/3/23 17:05	9,4	1199	45	34	2368	496	80	35	1199	36	25	2336	488	81	32
14/3/23 17:06	9,8	1200	44	33	2368	508	80	35	1200	41	26	2336	478	81	32
14/3/23 17:06	12	1501	55	58	2384	504	80	35	1500	55	60	2352	508	81	32
14/3/23 17:06	13,6	1500	56	60	2384	520	81	35	1499	56	60	2352	516	81	32
14/3/23 17:06	13,1	1500	55	61	2384	504	81	35	1498	56	65	2352	480	82	32
14/3/23 17:07	12,5	1500	53	60	2384	520	81	35	1499	56	70	2352	504	82	33
14/3/23 17:07	13	1500	54	60	2384	508	81	35	1498	56	75	2352	496	82	33
14/3/23 17:07	5,6	600	15	11	2320	212	78	35	602	16	15	2288	200	80	33
14/3/23 17:07	6,4	601	14	9	2320	204	78	35	601	15	14	2288	204	79	33
14/3/23 17:08	6,3	598	15	9	2320	220	78	35	599	15	14	2288	204	79	33
14/3/23 17:08	6,3	598	15	10	2320	220	78	35	600	15	14	2288	200	78	33
14/3/23 17:08	6,5	601	15	9	2320	252	78	35	601	15	15	2288	196	78	33
14/3/23 17:08	7,5	701	16	10	2336	254	78	36	700	19	15	2304	246	78	33
14/3/23 17:09	7,6	699	18	12	2336	272	78	36	700	19	15	2304	244	78	33
14/3/23 17:09	7	703	18	12	2336	272	78	36	695	20	15	2304	240	78	33
14/3/23 17:09	7,3	701	18	11	2336	276	78	36	700	20	15	2304	252	78	33
14/3/23 17:09	7,4	800	20	12	2336	300	78	36	797	22	20	2320	304	78	34
14/3/23 17:10	7,2	800	23	13	2336	312	78	36	798	22	19	2320	300	78	34
14/3/23 17:10	7,4	805	26	13	2336	326	78	37	796	23	19	2304	300	79	34
14/3/23 17:10	7,4	805	24	12	2336	312	78	37	804	23	19	2304	300	79	34
14/3/23 17:10	8	895	25	12	2352	326	78	37	892	26	20	2320	348	79	34
14/3/23 17:11	8,1	900	28	15	2352	320	78	37	897	24	20	2320	352	79	34
14/3/23 17:11	8	899	28	16	2352	326	78	37	900	25	20	2320	348	79	34
14/3/23 17:11	8	900	32	16	2352	368	78	37	889	25	20	2320	344	79	34
14/3/23 17:11	8,5	1002	32	17	2352	350	78	37	998	38	25	2320	416	79	34
14/3/23 17:12	8,2	1000	35	20	2352	404	79	37	1002	38	26	2336	398	79	34
14/3/23 17:12	8,5	998	34	20	2352	392	79	37	1003	38	24	2320	398	79	34
14/3/23 17:12	8,8	997	34	20	2352	404	79	37	999	39	28	2320	392	79	34
14/3/23 17:12	9	1200	36	24	2353	492	79	37	1196	50	32	2336	468	80	34
14/3/23 17:13	8,9	1198	40	29	2368	505	79	37	1194	50	32	2336	438	80	34
14/3/23 17:13	9	1199	45	30	2368	500	80	37	1195	50	31	2336	464	80	34
14/3/23 17:13	9	1198	49	35	2352	470	80	37	1197	48	32	2336	468	80	34
14/3/23 17:13	11	1499	48	60	2384	560	80	38	1500	56	70	2352	560	80	34
14/3/23 17:14	11,5	1499	56	60	2384	480	80	38	1501	57	75	2352	480	81	34
14/3/23 17:14	12,2	1498	56	65	2368	496	80	38	1498	55	82	2352	480	82	34
14/3/23 17:14	12,4	1499	56	70	2384	504	80	38	1465	54	74	2352	560	82	34