

E.T.S. de Ingeniería Industrial, Informática
y de Telecomunicación

Aprendizaje y planificación inteligente
de misiones de vuelo



Máster Universitario
en Ingeniería Informática

Trabajo Fin de Máster

Alumno: Jon Mikel Merino Ruiz

Director: Jesús Villadangos Alonso

Pamplona, 11 de junio de 2023

upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Palabras clave

- Dron/aeronave/UAV/UAS.
- Misión/operación.
- Plan de vuelo.
- Restricciones aéreas.
- Información geográfica/geoespacial/espacial.
- Información meteorológica e hidrológica.
- Cuenca hidrográfica.
- Capa vectorial.
- Pandas y geopandas.
- Camino más corto.

Símbolos y abreviaciones

- «UAV» Unmanned Aerial Vehicle
- «EMERAL» **EMER**ency **UAV**s for **L**ong Range Operations
- «EASA» Agencia de la Unión Europea para la Seguridad Aérea
- «AESA» Agencia Estatal de Seguridad Aérea
- «ATS» Servicios de Tránsito Aéreo
- «ENAIRE» Gestor de navegación aérea de España
- «IDENA» Infraestructura de Datos Espaciales de Navarra
- «INTA» Instituto Nacional de Técnica Aeroespacial

Índice general

Introducción y objetivos	7
1.1. Introducción	7
1.1.1. Proyecto EMERAL	7
1.2. Objetivos	8
Vehículos Aéreos no Tripulados	10
2.1. Componentes	11
2.2. Tipos de UAV	14
2.2.1. Clasificación por estructura	14
2.2.2. Clasificación por uso	15
2.2.3. Clasificación por normativa	16
2.3. Operaciones con UAS	18
2.3.1. Tipos de operaciones	18
2.3.2. Categorías operacionales	19
2.3.3. Operador de UAS	23
Planificación de vuelo	24
3.1. Servicios ATM/ANS	24
3.1.1. Organismos responsables de la Seguridad Aérea	25
3.1.2. Servicios para la Seguridad Aérea	26
3.2. Sistema UTM	29

3.2.1.	SESAR	31
3.2.2.	U-Space	31
3.3.	Planificación de vuelos autónomos	34
3.4.	Partes de un plan de vuelo	35
3.4.1.	Meteorología	36
3.4.2.	Entorno	36
3.4.3.	UAS	38
3.4.4.	Limitaciones Humanas	38
Servicio de información meteorológica		39
4.1.	Conjunto de datos	39
4.2.	Fuentes de datos	41
4.2.1.	IDENA	41
4.2.2.	AEMET	43
4.2.3.	Meteo Navarra	43
4.2.4.	OpenData Navarra	45
4.3.	Procesamiento de datos	45
4.4.	Despliegue	48
Identificación de restricciones de vuelo		50
5.1.	Restricciones de vuelo de drones	50
5.1.1.	Capas de restricciones	51
5.1.2.	Fuente de información de ENAIRE	53
5.1.3.	Descarga de datos	54
5.1.4.	Gestión de restricciones	55
5.2.	Despliegue	57
5.2.1.	API Comunicación Tesicnor e i3Code	57
5.2.2.	Web EMERAL	60
Generación de la ruta de vuelo		62

6.1. Contexto	62
6.2. Datos red hidrográfica	62
6.2.1. Obtención de datos	63
6.2.2. Preprocesamiento de datos	64
6.3. Proceso de generación de la ruta de vuelo	67
6.3.1. Descripción general del proceso	67
6.3.2. Búsqueda del punto más cercano	68
6.3.3. Búsqueda del camino más corto	69
6.4. Despliegue	81
Conclusiones y líneas futuras	85
7.1. Conclusiones	85
7.2. Líneas futuras	86
Bibliografía	87

Índice de figuras

1.	Consortio proyecto EMERAL	8
2.	Componentes del dron	11
3.	Tipos de drones según su estructura	14
4.	Agencias de Seguridad Aérea	25
5.	Proveedores de Servicios de Seguridad Aérea	26
6.	Servicios ATM/ANS	26
7.	Comparativa servicios ATM y UTM	30
8.	Fases U-Space	32
9.	Partes Plan de Vuelo	35
10.	Restricciones de vuelo ENAIRE	37
11.	E/R datos meteorológicos y de aforo	39
12.	Fuentes de información	41
13.	Descarga de la capa Municipios	42
14.	Estación Meteorológica UPNA	44
15.	Capas Municipios Cauces y Estaciones	47
16.	Métodos Servicio Meteorológico e Hidrológico	48
17.	Métodos Servicio Meteorológico e Hidrológico	49
18.	Listado capas ENAIRE	51
19.	Tarea programada de Windows descarga datos ENAIRE	55
20.	Servicio ArcGIS	56
21.	Diagrama secuencia método <i>Generate</i>	59

22.	Listado vuelos	60
23.	Plan de vuelo río Irati	61
24.	Capas ríos España	63
25.	Tramos del río Bidasoa	65
26.	Procesamiento geometría río Bidasoa	66
27.	Punto más cercano del río Salazar	68
28.	Bounding Box zona Lumbier	70
29.	Ríos y puntos de intersección	71
30.	Grafo ríos zona Lumbier	72
31.	Grafo ponderado zona Lumbier	75
32.	Camino más corto en el grafo	79
33.	Camino más corto ejemplo Lumbier	79
34.	Ejemplo río trenzado (Río Cinca)	80
35.	Mapa interactivo web EMERAL	81
36.	Pasos generación automática de la ruta	82
37.	Ruta por río Irati desde Aos hasta Murillo de Lónguida	83
38.	Ruta por río Irati desde Aos hasta Artajo	83

Introducción y objetivos

1.1. Introducción

Los drones cada vez están cobrando más relevancia, en el futuro se prevé que formarán parte de nuestra vida cotidiana, tal y como describe Bartolomé Marqués Balaguer, responsable de la Subdirección General de Sistemas Aeronáuticos de INTA, en el artículo [1]. Actualmente, el uso de drones cobra importancia en situaciones de emergencia, como por ejemplo para llegar a zonas de difícil acceso azotadas por desastres naturales. Además, su velocidad de vuelo permite recorrer largas distancias en poco tiempo. En esta situación se encuentra enmarcado el proyecto EMERAL, proyecto incluido en la convocatoria de Proyectos Estratégicos de I+D 2021 del Gobierno de Navarra.

1.1.1. Proyecto EMERAL

El objetivo del proyecto EMERAL es proveer a los servicios de emergencia de herramientas de gestión integrada e inteligente de los datos suministrados por drones antes, durante y después de una inundación. Esta necesidad ha sido detectada por Tesicnor, empresa coordinadora del proyecto, la cual ha desarrollado la aplicación NOÉ, sistemas de gestión de inundaciones que se encuentra ya en el mercado. El producto final del proyecto EMERAL es la implementación del sistema NOÉ+ correspondiente al actual sistema NOÉ con la integración de drones de largo alcance.

El Ministerio para la Transición Ecológica y el Reto Demográfico invitó a Tesicnor

a la jornada anual sobre I+D+i en la gestión de inundaciones, celebrada en Escuela Nacional de Protección Civil el miércoles 29 de marzo de 2023, y en la que Tesicnor tubo la oportunidad de presentar el sistema NOÉ y su versión mejorada NOÉ+.



Figura 1: Consorcio proyecto EMERAL

Para resolver las necesidades del proyecto, este está formado por un consorcio que cuenta con toda la cadena de valor de la solución, tal y como vemos en la Figura 1. Nosotros nos situamos en la empresa i3Code, encargada del desarrollo de soluciones que ayuden a los pilotos de drones en la planificación del vuelo, la cuál realizan en base a distintos factores (meteorológicos, geográficos, aéreos, legislativos etc.). El objetivo principal de este trabajo automatizar parte de esta planificación. Por una parte, vamos a identificar las restricciones de vuelo definidas por ENAIRE, que se encuentran en la zona de vuelo. Más adelante veremos qué tipo de restricciones existen y a qué vuelos afectan. Por otra parte, vamos a agilizar la generación de los puntos o waypoints que definen la trayectoria del dron, automatizando este proceso para vuelos realizados sobre corrientes de agua superficiales. Es decir, el algoritmo de planificación de vuelo que vamos a implementar va a ser capaz de identificar la línea de puntos sobre el cauce de un río.

1.2. Objetivos

- Generar script de la misión de vuelo con metadatos que aporten al piloto información relevante sobre el vuelo.
- Identificar las restricciones de vuelo definidas por ENAIRE que afectan a una misión de vuelo.

- Generar de forma automática los puntos o waypoints a lo largo de corrientes de agua superficiales.
- Implementar una aplicación web que permita a los usuarios la realización de los objetivos anteriores.

Vehículos Aéreos no Tripulados

Los Vehículos Aéreos no Tripulados o *Unmanned Aerial Vehicle* UAV, son aeronaves que se controlan de forma remota y que vuelan sin tripulación, es decir, el piloto de la aeronave no se encuentra a bordo de la misma. Los UAVs son comúnmente conocidos como drones, aunque también se emplean otros términos con significados similares. El término Aeronave Pilotada por Control Remoto o *Remotely Piloted Aircraft* RPA, como su propio nombre indica, hace referencia a aeronaves controladas por un piloto de manera remota, descartando los drones programables para realizar misiones de vuelo de forma autónoma. Por lo tanto, tenemos que los RPAs son un subgrupo de los UAVs. Por otra parte, tenemos los términos Sistemas Aéreos no Tripulados o *Unmanned Aerial System* UAS, y Sistemas de Aeronaves Pilotadas por Control Remoto o *Remotely Piloted Aircraft System* RPAS, que hacen referencia al sistema completo de vuelo de un UAV y de un RPA, respectivamente. Estos sistemas engloban la aeronave, el enlace de comunicaciones y la estación en tierra desde donde se controla al dron. Al igual que con los términos anteriores, los RPASs son un subgrupo de los UASs. En este trabajo descartamos los RPAs y RPASs ya que los objetivos del mismo implican que los drones presenten la capacidad de realizar vuelos programados.

En este capítulo veremos los componentes de los que están formados los UAVs, así como su clasificación en base a sus usos y características. También entraremos en la parte legislativa que regula el uso de drones. En concreto, veremos las clases de drones y las categorías operacionales recogidas en la normativa europea.

2.1. Componentes



Figura 2: Componentes del dron

En la Figura 2 podemos observar los componentes principales de un dron.

1. Estructura:

La estructura del dron, también conocida como marco, es la pieza base dónde se montan el resto de componentes. Está diseñado para reducir las vibraciones que provoca el motor al girar las hélices. Según la estructura, podemos clasificar a los drones en dos tipos: de ala fija o de ala rotatoria. En la Subsección 2.2.1 explicaremos las diferencias entre ambos.

2. Batería:

La batería alimenta a los componentes eléctricos del dron. La autonomía del mismo dependerá del consumo de energía que requiere, así como de la calidad de la batería. Entre los drones de uso comercial, esta va desde unos 5-10 minutos en los drones más básicos, hasta los 25-30 minutos en los drones de gama alta.

3. Motores:

El motor o motores de los drones mueven sus hélices para elevarlo del suelo y mantenerlo en el aire. La mayoría de motores son eléctricos, normalmente usados

en drones de menor dimensión, debido a sus bajos consumos de energía. También se emplean motores de gasolina, normalmente en drones de mayor dimensión que requieren más potencia y autonomía.

4. Cámara:

La cámara captura imágenes y videos durante el vuelo. Es una componente opcional, aunque resulta interesante para sacar el máximo potencial a los drones, ya que permiten la toma de imágenes aéreas desde perspectivas que no se pueden tomar con ningún otro mecanismo.

5. Sensores:

Los drones tienen una serie de sensores para controlar el vuelo. La placa controladora de vuelo recoge todos los datos de los sensores y ordena el movimiento del dron según las indicaciones que recibe del control remoto. Los sensores más importantes son:

- **Sensores de posición:** La posición del dron se puede determinar con distintos sistemas o una combinación de ellos. El más empleado es el Sistema de Posicionamiento Global o *Global Positioning System* GPS, que es un sistema de navegación que recoge información satelital para determinar la posición del dron de manera precisa y a tiempo real.
- **Sensores de altitud y altura:** La altitud y la altura son términos distintos, la primera es la distancia del dron respecto al nivel del mar y la segunda es la distancia a la que se sitúa el dron del suelo. El barómetro es el instrumento clásico que determina la altitud, compara la presión atmosférica que rodea al dron con la presión al nivel del mar. En cambio, el instrumento clásico para determinar la altura es el radioaltímetro, emite pulsos de ondas electromagnéticas hacia la superficie que sobrevuela la aeronave. Debido al principio de la reflexión de estas ondas electromagnéticas, es posible calcular el tiempo que tarda la señal entre su emisión y recepción y por lo tanto la altura relativa

a la que se encuentra el dron.

- **Sensores de control de vuelo:** La actitud que adopta un dron en el vuelo depende de distintos factores, tales como el propósito del vuelo o factores ambientales del entorno. Por lo tanto, la actitud de un dron se ajusta y calibra antes de cada vuelo para garantizar un rendimiento óptimo y una operación segura. Para ello, los drones cuentan con una Unidad de Medición Inercial o *Inertial Measurement Unit* IMU, formada por una combinación de acelerómetros, giroscopios y magnetómetros para determinar la orientación, posición y velocidad del dron.
 - El acelerómetro mide la aceleración del dron en tres dimensiones (x, y, z), permitiéndole ajustar su posición y orientación en respuesta a los cambios en el entorno.
 - El giroscopio mide la velocidad angular de rotación de un dron, permitiéndole mantener su orientación en el aire.
 - El magnetómetro mide tanto la fuerza como la dirección de los campos magnéticos, permitiendo al dron calcular su rumbo y orientación.

Los datos de la IMU son procesados por el controlador de vuelo del dron, que utiliza la información para estabilizar el vuelo, incluso en entornos difíciles o condiciones de mucho viento.

6. Hélices:

Los drones de ala rotatoria poseen hélices, cuyo número dependerá del número de rotores. Su funcionamiento es semejante al de un helicóptero, el motor hace girar las hélices para elevar y mantener al dron en el aire.

7. Gimbal:

El gimbal es un estabilizador de la cámara del dron, el cual logra que la calidad de las imágenes no se vea afectada, incluso cuando el dron experimenta turbulencias.

2.2. Tipos de UAV

El sector de los UAV ha crecido mucho en los últimos años, por lo que cada vez encontramos distintos tipos de drones en función de nuevos uso y características.

2.2.1. Clasificación por estructura

Existen dos tipos de drones según su estructura o forma:



(a) Dron ala fija

(b) Dron de ala rotatoria

Figura 3: Tipos de drones según su estructura

- Drones de ala fija:

Los drones de ala fija tienen un aspecto similar a un avión, en dónde sus alas son las que generan la fuerza de sustentación en el aire. Podemos ver un ejemplo en la Figura 3(a), corresponde a un modelo desarrollado por Fuvex, empresa situada en Tudela (Navarra) que proporciona servicios de integración de drones. En concreto, este dron se ha empleado para la vigilancia del tendido eléctrico a vista de pájaro, labor que se realizaba mediante helicóptero, medio que supone un mayor coste, una gran contaminación y un peligro para vidas humanas. En el vídeo [2] se muestra cómo el dron realiza dicha inspección de la red eléctrica.

- Drones de ala rotatoria:

Los drones de ala rotatoria o multirrotores se mantienen en el aire por medio de las hélices ubicadas en cada uno de sus brazos. Son los más comunes en el merca-

do actual por su versatilidad, los cuales logran mantener una posición estática en el aire gracias a sus componentes electrónicos como giroscopios o estabilizadores. Según la cantidad de motores o brazos que poseen, podemos distinguir entre tricópteros (3), cuadricópteros (4), hexacópteros (6) o octocópteros (8). El número de brazos es proporcional a la estabilidad que ofrece el dron. En la Figura 3(b) se muestra un prototipo de hexacóptero desarrollado por la Universidad Pública de Navarra. También existe otro tipo de drones denominado coaxial, los cuales tienen dos motores en cada brazo, ofreciendo así una potencia que les permite transportar objetos pesados.

A lo largo del trabajo nos centraremos en los drones de ala rotatoria, ya que se ajustan mejor a los objetivos del mismo.

2.2.2. Clasificación por uso

Según las características de los drones, pueden ser usados con distintas finalidad. Los usos más comunes son los siguientes:

- **Recreativo:**

Los drones con fines recreativos son de uso particular, empleados para entretener a sus propietarios.

- **Comercial:**

Los drones con fines comerciales son empleados para multitud de usos, tales como grabaciones en eventos de distinta índole o tomas aéreas de paisajes.

- **Geográficos:**

En el ámbito geográfico, son útiles en el área de la topografía, más concretamente en las ramas de planimetría y altimetría, para realizar mediciones desde una posición de altura. La planimetría trata de representar el terreno a escala en una superficie plana. Por su parte, la altimetría trata de representar el relieve del terreno.

- Agrónomos:

En el campo de la agronomía, se utilizan frecuentemente para el cuidado de cultivos, minimizando así la intervención de mano de obra en los campos gracias a la fumigación con drones.

- De transporte y movilidad:

En algunos lugares los drones son usados para hacer envíos de mercancías, aunque las regulaciones para este tipo de actividad son bastante rigurosas.

- De seguridad:

Como ya se ha detallado al inicio del trabajo, actualmente el uso de drones está cobrando importancia en el ámbito de la seguridad. Un ejemplo de ello son los drones de vigilancia, los cuales capturan imágenes en zonas que requieren vigilancia convencional, pero que el uso de UAV les aporta una mayor capacidad de portabilidad, rapidez y usabilidad. Por ejemplo con el uso de cámaras térmicas para tomar imágenes nocturnas. Estas imágenes se previsualizan de forma automática y en tiempo real en el puesto de mando.

2.2.3. Clasificación por normativa

De acuerdo a la normativa europea vigente, existen 7 clases de UAS que van desde la clase C0 a la clase C6. Las 5 primeras se establecen como Anexo en el Reglamento [3] y las dos últimas se incluyen en la modificación [4] del Reglamento anterior. La clasificación se realiza a partir de una serie de características del dron, tales como:

- MMD: Masa máxima al despegue.

También conocida con las siglas MTOM (*Maximun Take-Off Weight*).

- VM: Velocidad máxima en vuelo horizontal.

- AM: Altura máxima desde el punto de despegue.

Clase	MMD	VM	AM	Otras características
C0	<250g	19m/s	120m	<ul style="list-style-type: none"> - Controlarse de manera segura. - Diseño que reduzca al máximo las lesiones de personas. - Equipado con un modo sígueme. - Comercializado con un manual de usuario.
C1	<900g	19m/s	120m	<ul style="list-style-type: none"> - Características clase C0. - Energía de impacto menor de 80 J. - Resistencia mecánica para evitar roturas o deformaciones. - Método de recuperación de conexión o finalización de vuelo de forma segura. en caso de pérdida de señal. - Número de serie físico único. - Alimentado exclusivamente con electricidad. - Sistema de identificación a distancia que envíe información del vuelo a tiempo real. - Sistema de geoconsciencia que evite la violación de los límites del espacio aéreo. - Señal de alerta cuando se alcance un nivel bajo de batería. - Equipado con luces que permita la visibilidad a la noche.
C2	<4kg	-	120m	<ul style="list-style-type: none"> - Características clase C1, excepto el modo sígueme. - Anclaje con una longitud de tracción inferior a 50 m (drones cautivos). - Enlace de datos protegido contra el acceso no autorizado a las funciones de control. - Modo de baja velocidad con un límite máximo de 3 m/s, excepto drones de ala fija.
C3	<25kg	-	120m	<ul style="list-style-type: none"> - Características clase C2, excepto el modo de baja velocidad. - Dimensión característica máxima inferior a 3 m.
C4	<25kg	-	-	<ul style="list-style-type: none"> - Controlarse de manera segura. - No disponer de modos de control automático, excepto para estabilizar el vuelo o en casos de pérdida de la conexión. - Comercializado con un manual de usuario.
C5	<25kg	-	-	<ul style="list-style-type: none"> - Características clase C3, con el sistema de geoconsciencia opcional. - No tener estructura de ala fija, excepto drones cautivos. - Modo de baja velocidad con un límite máximo de 5 m/s.
C6	<25kg	50m/s	-	<ul style="list-style-type: none"> - Características clase C3, con el sistema de geoconsciencia opcional y exceptuando la alimentación exclusiva con electricidad. - Proporcionar medios para programar la trayectoria.

Tabla 2.1: Clases de drones de acuerdo a la legislación actual

En las clases C2, C3 y C5 de la Tabla 2.1, se determinan ciertas especificaciones técnicas para los drones cautivos. Los drones cautivos son aquellos que poseen un sistema de alimentación desde una fuente de energía situada en tierra, mediante un cable eléctrico que conecta el dron a la estación base. Este sistema supone una gran ventaja en cuanto a la autonomía del dron, permitiendo mantener el vuelo durante horas e incluso días, respecto a los 20 minutos de media que duran las baterías.

2.3. Operaciones con UAS

Las operaciones con UAS tienen como objetivo regularizar el uso de drones dentro de los diferentes espacios aéreos, garantizando la seguridad de los sistemas de aviación existentes. El 31 de diciembre de 2020 entró en vigor la normativa europea de UAS, elaborada por la EASA y recogida en los Reglamentos [3] y [5]. Esta normativa, afecta a todos los drones independientemente de su uso o tamaño. En la Subsección 3.1.1, veremos las competencias de EASA y de su homóloga a nivel nacional (AESA).

2.3.1. Tipos de operaciones

Existen 3 variaciones de operaciones con drones, que se diferencian principalmente por la distancia máxima a la que puede volar un dron desde la ubicación del piloto.

VLOS

Las operaciones VLOS (*Visual Line of Sight*) son las que se realizan dentro del alcance visual del piloto, es decir, el piloto debe visualizar el dron durante todo el vuelo. El piloto debe tener en cuenta los obstáculos que puedan entorpecer la visión, tales como la niebla, la falta de luz o el deslumbramiento causado por el sol. Por lo general, el dron no podrá estar a más de 500 metros ni a más de 120 metros de altura respecto al punto más alto en un radio de 150 metros.

EVLOS

Las operaciones EVLOS (*Extended Visual Line of Sight*) son una versión extendida de las operaciones VLOS, en dónde una o varias personas mantienen la visualización con el dron y una comunicación con el piloto mediante un dispositivo de radio, para informarle de la actitud que toma el dron durante el vuelo. Además, estas personas deben tener licencia de pilotos profesionales de drones. El radio en el que se puede mover el dron aumenta en 500 metros por cada observador adicional.

BVLOS

Las operaciones BVLOS (*Beyond Visual Line Of Sight*) son las que superan el alcance visual con el piloto. En este caso, la monitorización del dron se realiza desde una estación de control remoto. Además, es necesario que el dron pase por un estudio de seguridad y disponga de un dispositivo de visión dirigido hacia delante. En los casos en los que los vuelos BVLOS no se realicen en zonas específicamente delimitadas para este tipo de vuelo, el dron debe estar equipado con un sistema de detección de otras aeronaves y de personas. Estas operaciones presentan algunas limitaciones:

- Se requiere de la solicitud de publicación de un NOTAM, los cuales son un tipo de alertas que en el capítulo “Planificación de vuelo” veremos con más detalle.
- No se pueden realizar sobre núcleos de población o zonas de aglomeración de personas.
- El dron debe tener un sistema de enlace óptimo con la estación base y nunca superar los límites de seguridad de conexión.
- El vuelo deberá realizarse solo por pilotos de dron profesionales y acreditados de acuerdo a las exigencias de la AESA.

2.3.2. Categorías operacionales

Las operaciones con UAS se realizan en las categorías abierta, específica o certificada, definidas en los artículos 4, 5 y 6 del Reglamento [5], respectivamente.

Categoría abierta

La categoría abierta incluye todas las operaciones de bajo riesgo operacional. Por lo que no requieren autorización de AESA ni una declaración responsable del operador de UAS. Los requisitos que deben cumplir la operación en la categoría abierta son:

- El UAV pertenece a alguna clase de la Tabla 2.1 o es de construcción privada.

- La MMD <25kg.
- El UAV no sobrevuela concentraciones de personas y se mantiene a una distancia segura de las personas.
- Operaciones de tipo VLOS o EVLOS.
- La UAV no puede superar los 120 m de altura respecto al punto más próximo de la superficie, excepto para superar un obstáculo artificial de una altura superior a 105 m, en cuyo caso se permite aumentar en 15 m el límite de 120 m.
- No se transportan mercancías peligrosas ni se dejan caer objetos desde el dron.

La categoría abierta se divide en tres subcategorías:

- Subcategoría A1

Las operaciones de la subcategoría A1 deben realizarse con drones de las clases C0 o C1. Con drones de la clase C0 no se puede sobrevolar por encima de personas que no participan en la operación, y se debe evitar hacerlo cuando se emplean drones de la clase C1.

Los pilotos tienen que estar familiarizados con el manual de usuario dispuesto por el fabricante. Además, en caso de drones de clase C1, el piloto debe completar un curso y aprobar un examen teórico de forma online, realizado por una autoridad competente.

- Subcategoría A2

Las operaciones de la subcategoría A2 deben realizarse con drones de las clase C2. Se debe mantener una distancia horizontal de seguridad de al menos 30 m respecto a las personas que no participan en la operación. Aunque se puede reducir esa distancia hasta 5 m si el piloto activa del modo de baja velocidad, siempre y cuando las condiciones lo permitan. En este caso los pilotos también tienen que superar un examen teórico en línea, además de realizar un formación de autopráctica.

- Subcategoría A3

Las operaciones de la subcategoría A3 deben realizarse con drones de las clases C2, C3 o C4, en una zona donde no se ponga en peligro a ninguna persona y a una distancia horizontal mínima de 150 m de zonas residenciales, comerciales, industriales o recreativas. En este caso los pilotos también tienen que superar un examen teórico en línea.

Categoría específica

La categoría específica incluye todas las operaciones de riesgo medio que no pueden realizarse en categoría abierta. En España, es necesario estar habilitados como operadores UAS ante AESA. Además, el operador UAS deberá solicitar una autorización operacional de acuerdo con el artículo 12 del Reglamento [5]. Esta solicitud deberá ir acompañada de una serie de medidas de atenuación del riesgo que supone la operación.

Dentro de la categoría específica, existen dos escenarios estándar (STS) definidos la autoridad competente, en los que ya se ha realizado el estudio de seguridad y se ha definido las medidas de atenuación que el operador debe aplicar. Por lo tanto, el operador UAS no estará obligado a obtener una autorización operacional cuando la operación que se ajuste a un STS. En su lugar, podrá presentar una declaración operacional de conformidad con un STS.

Hasta el 31 de diciembre de 2023, esta declaración operacional se debe realizar sobre los STS nacionales (STS-ES) recogidos en [6]. A partir del 1 de enero de 2024, la declaración operacional se debe realizar sobre los STS europeos recogidos en el Reglamento [5].

- STS nacionales

- STS-ES-01: Operaciones VLOS sobre una zona terrestre controlada en un entorno poblado.
- STS-ES-02: Operaciones BVLOS con observadores del espacio aéreo sobre una zona terrestre controlada en un entorno poco poblado.

- STS europeos
 - STS-01: Operaciones VLOS sobre una zona terrestre controlada en entorno poblado, con drones de la clase C5.
 - STS-02: Operaciones BVLOS sobre una zona terrestre controlada en un entorno escasamente poblado, con drones de la clase C6.

El Reglamento [5] define “zona terrestre controlada” como la zona terrestre en la que se utiliza el UAS y en la que el operador de UAS puede garantizar que solo estén presentes las personas participantes.

En los STS nacionales no se especifica la clase a la que pertenece la aeronave, pero sí que se definen una serie de características que debe tener. No puede ser de ala fija, debe pesar menos de 10kg, ser eléctrica, estar configurada a 5 m/s de velocidad máxima, incluir medios para reducir el efecto de la dinámica de impacto y disponer de un seguro.

Categoría certificada

La categoría certificada incluye las operaciones de alto riesgo operacional. Por lo que se requiere certificación del UAS por la autoridad competente, la certificación del operador y, en su caso, la obtención de una licencia por parte del piloto. El Reglamento [5] estipula los siguientes requisitos que deben cumplir las operaciones para clasificarse en la categoría certificada:

- Condiciones de la aeronave:
 - Dimensión característica de 3 m o más y diseño para ser utilizado sobre concentraciones de personas.
 - Diseño para el transporte de personas.
 - Diseño para el transporte de mercancías peligrosas y requiere una gran solidez para atenuar los riesgos para terceros en caso de accidente.
- Condiciones de la operación:

- Implica volar sobre concentraciones de personas.
- Conlleva el transporte de personas.
- Conlleva el transporte de mercancías peligrosas que pueden entrañar un riesgo elevado para terceros en caso de accidente.

Además, las operaciones de UAS se clasificarán en la categoría certificada si la autoridad competente así lo considera tras evaluar el riesgo de la operación en base al artículo 11 del Reglamento [5].

Actualmente, la categoría certificada se encuentra en desarrollo a nivel europeo.

2.3.3. Operador de UAS

Según el Reglamento [5], un operador de UAS incluye a toda persona física o jurídica que utilice o tenga intención de utilizar uno o varios UAS, tanto para vuelos profesionales como recreativos. Los operadores pueden estar formados por equipos interdisciplinarios o por un solo piloto. En España, el registro como operador de UAS se realiza en AESA, siempre y cuando se realicen operaciones con los siguientes tipos de UAS:

- En categoría abierta, con un dron de MMD de al menos 250g y con una energía de impacto superior a 80J. Si MMD es menor de 250g, es obligatorio el registro como operador de UAS si el dron es capaz de capturar datos personales.
- En categoría específica con drones de cualquier masa.

Planificación de vuelo

Los pilotos de UAVs, realizan operaciones en base a planes de vuelos, en los que se predefine una serie de parámetros que afecta de forma directa o indirecta a la operación que se va a llevar a cabo. Algunos de estos parámetros son la ruta que sigue el dron, las condiciones del lugar de vuelo, las personas participantes de la operación y objetivo de la misma. La parte correspondiente a la definición de la zona de vuelo se debe realizar en base a los servicios de tránsito aéreo. Resulta de vital importancia el control sobre el espacio aéreo donde se vaya a realizar la operación, con el fin de mantener la seguridad, sobretodo en las cercanías de las afluencias de aeropuertos, aeródromos o helipuertos.

A continuación explicaremos qué organismos públicos se encargan de mantener la seguridad aérea, y qué servicios ofrecen para cumplir con ese objetivo. También veremos la importancia de los vuelos programados y cómo realizarlos, así como las partes más importantes de un plan de vuelo.

3.1. Servicios ATM/ANS

La Gestión de Tránsito Aéreo o *Air Traffic Management* (ATM) y los Servicios de Navegación Aérea o *Air Navigation Services* (ANS), prestan una serie de servicios con el fin de que las operaciones aéreas se realicen con seguridad. Antes de concretar cuáles son estos servicios, veremos los organismos públicos encargados de regularlos y de garantizar su aplicación.

3.1.1. Organismos responsables de la Seguridad Aérea



Figura 4: Agencias de Seguridad Aérea

En la Figura 4 podemos observar los organismos públicos responsables de la seguridad aérea, tanto a nivel europeo (EASA), como a nivel nacional (AESA).

La Agencia Europea de Seguridad Aérea o *European Aviation Safety Agency* (EASA), asesora técnicamente a la Comisión Europea (CE) sobre cuestiones de aviación. La CE es la encargada de redactar la regulación sobre seguridad de la aviación dentro del espacio aéreo de los Estados miembros, así como de garantizar la aplicación uniforme a través de inspecciones a las Autoridades de Nacionales de Aviación Civil de cada país de la Unión Europea. Estas son las encargadas de la mayoría de tareas operativas, como la expedición de certificaciones al personal aeronáutico o de autorizaciones operacionales para realizar vuelos.

En España, la Agencia Estatal de Seguridad Aérea (AESA) es un organismo adscrito a la Secretaría de Estado de Transporte del Ministerio de Transportes, Movilidad y Agenda Urbana, que se encarga del cumplimiento de la normativa de aviación civil en el conjunto de la actividad aeronáutica de España. En caso de incumplimiento, AESA tiene la potestad sancionadora.



Figura 5: Proveedores de Servicios de Seguridad Aérea

Entre las competencias de AESA, también se encuentra la supervisión de una serie de servicios necesarios para la gestión segura de las operaciones aéreas. En la Figura 5 podemos observar las empresas certificadas por AESA como proveedores de estos servicios. Estas certificaciones se encuentran recogidas en [7], dónde también podemos ver qué servicios prestan cada una de las empresas. Estas empresas, pueden ser públicas, entre las que destaca ENAIRE, o privadas, entre las que destaca SAERCO.

3.1.2. Servicios para la Seguridad Aérea

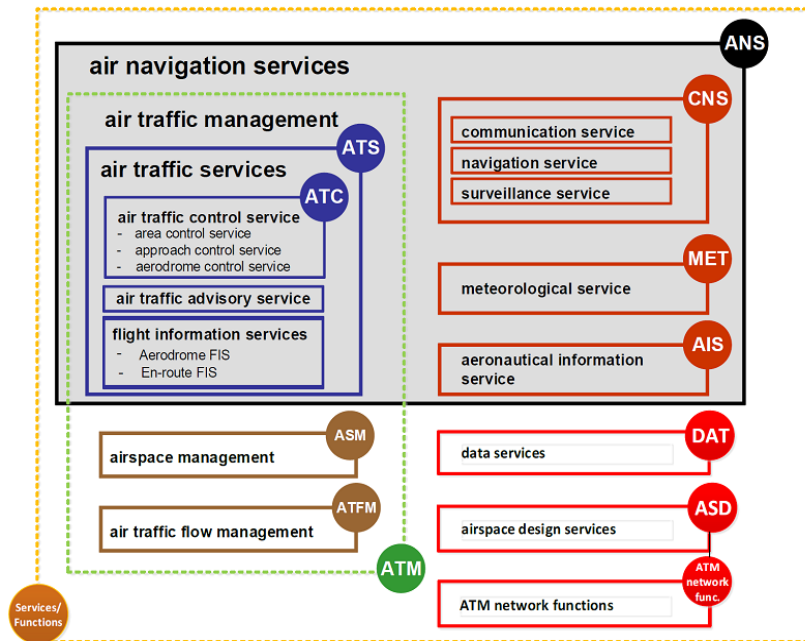


Figura 6: Servicios ATM/ANS

En la Figura 6 podemos observar el conjunto de los servicios ATM/ANS. Vamos a explicar brevemente cada uno de ellos:

Servicios de Tránsito Aéreo

Los Servicios de Tránsito Aéreo o *Air Traffic Service* ATS, regulan y asisten a las aeronaves, con el fin de garantizar que las operaciones aéreas sean seguras, regulares, económicas y eficientes.

Los ATS están formados por 3 servicios:

- Servicio de Control de Tránsito Aéreo

El Servicio de Control de Tránsito Aéreo o *Air Traffic Control Service* ATC, previene colisiones entre aeronaves que se encuentran realizando un vuelo y entre las aeronaves y los obstáculos que se encuentran en el espacio aéreo controlado. También acelera y mantiene el orden en el tráfico aéreo. Los ATC están formados por 3 servicios prestados por controladores aéreos y en los que se emplean dos infraestructuras situadas en los aeropuertos: la Torre de Control (TWR) y el Centro de Control de Área (ACC).

- Servicio de Control de Área: El ACC controla las aeronaves en ruta.
- Servicio de Control de Aproximación: La TWR o el ACC controlan la llegada y salida de aeronaves a un Aeródromo mientras están en vuelo.
- Servicio de Control de Aeródromo: La TWR controla el movimiento de las aeronaves que están en la tierra y las que aterrizan, despegan y vuelan en las cercanías.

- Servicio de Información de Vuelo

El Servicio de Información de Vuelo o *Flight Information Service* FIS, asesora y proporciona información útil para la marcha segura y eficaz de los vuelos. El FIS aporta cualquier información que pueda afectar a la seguridad operacional. El Servicio de Información de Vuelo de Aeródromos o *Flight Information Service*

AFIS, se establece en aquellos aeródromos que no requieren un servicio ATC, pero sí que es necesario proveer de información a las aeronaves en el área de responsabilidad del aeródromo, para ayudar a los pilotos a evitar colisiones.

- Servicio de Alerta

El Servicio de Alerta o *Air Traffic Advisory Service* ATAS, auxilia y/o notifica a las dependencias pertinentes respecto a las aeronaves que se encuentran en situación de emergencia.

Servicio Comunicaciones-Navegación-Vigilancia

El Servicio Comunicaciones-Navegación-Vigilancia o *Communication-Navigation-Surveillance Service* CNS, da soporte a la navegación y la circulación aérea gracias al conocimiento de la posición de la aeronave y el cumplimiento de su plan de vuelo de forma segura y eficiente sin entrar en conflicto con otras aeronaves.

Se compone de tres pilares:

- Comunicaciones: Proporciona soporte a las comunicaciones Tierra/Tierra, Tierra/Aire y Tierra/Espacio entre dependencias de control y aeronaves.
- Navegación: Proporciona a las aeronaves información sobre el espacio y el tiempo en el que se encuentran.
- Vigilancia: Determina las posiciones relativas de las aeronaves para establecer distancias seguras de separación entre las mismas gracias a que proporciona al controlador posición, identificación, altitud y actitud de las aeronaves en ruta o aproximación bajo su jurisdicción.

Servicio de Meteorología Aeronáutica

Servicio de Meteorología Aeronáutica o *Meteorological Service* MET, difunde información meteorológica en tiempo real y realiza pronósticos de condiciones que peligran las operaciones.

Servicio de Información Aeronáutica

El Servicio de Información Aeronáutica o *Aeronautical Information Service* AIS, proporciona la información aeronáutica necesaria para la seguridad operacional, regularidad y eficiencia de la navegación aérea.

Existe una estandarización para el uso operacional de esta información aeronáutica por parte de la aviación civil internacional. La Organización de Aviación Civil Internacional (OACI) establece la forma en el que los AIS recogen y gestionan esta información. En España, el único organismo responsable de la provisión de este servicio como proveedor certificado AIS es ENAIRE.

Gestión del Espacio Aéreo

La Gestión del Espacio Aéreo o *Airspace Management* ASM, maximiza la utilización del espacio aéreo disponible por parte de sus múltiples usuarios, civiles y militares, de la forma más eficiente y optimizada posible.

Gestión de Afluencia del Tráfico Aéreo

La Gestión de Afluencia del Tráfico Aéreo o *Air Traffic Flow Management* ATFM, contribuye a un flujo seguro y ordenado del tráfico aéreo, asegurando que la capacidad del ATC se utiliza en la mayor medida posible, y que el volumen de tránsito es compatible con las capacidades declaradas por la autoridad de los servicios del tránsito aéreo.

3.2. Sistema UTM

El tradicional ATM no es compatible con los drones ya que, como hemos visto, presta servicios basados en la comunicación con los pilotos. El rápido crecimiento del sector UAS, tal y como comentamos Sección 1.1, requiere desarrollar los sistemas ATM con un enfoque especializado para la Gestión del Tráfico Aéreo no Tripulado o *Unmanned Traffic Management* UTM. Esta gestión incluye el conjunto de métodos, servicios y

herramientas con los que integrar y coordinar el tráfico aéreo de UAS con el de aeronaves tripuladas, de forma segura, ordenada y eficiente.

UTM, debe verse como una componente más de ATM y no como un servicio complementario. No se debe considerar el sistema UTM como un forma de segregar un espacio aéreo para el uso exclusivo de los UAS, sino como un entorno en el que se pueda garantizar un buen uso del espacio aéreo por parte de toda la aviación.

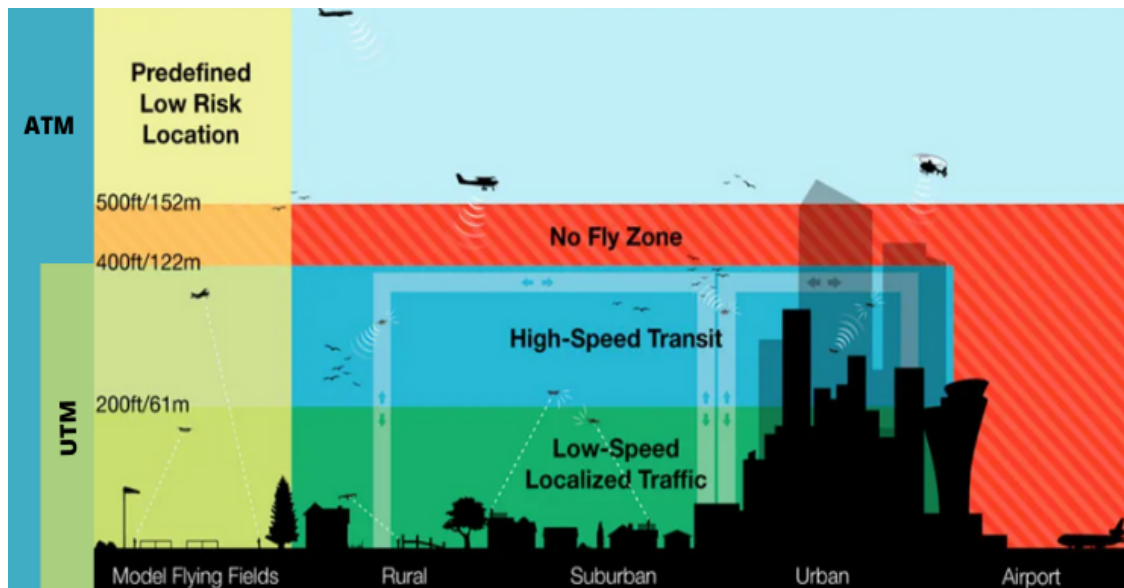


Figura 7: Comparativa servicios ATM y UTM

Como vemos en la Figura 7, el sistema UTM establece el vuelo de UAS a una altitud máxima de unos 120 metros (400 pies), tanto en zonas rurales como en urbanas.

Además, se debe asegurar la comunicación bidireccional entre la aeronave y los sistemas UTM, de manera que se tenga acceso en tiempo real a información relativa a la operación, tal como la posición de la aeronave, el plan de vuelo que está siguiendo, condiciones meteorológicas, etc. Esta información va a permitir a la autoridad aérea del país publicar restricciones previas a la operación y/o ocasionadas durante la misma, siendo los operadores de UAS los responsables de cumplir con dichas restricciones. La intervención de los operadores solo será necesaria en casos de emergencia, ya que UTM

está diseñado para usarse sobretodo para operaciones BVLOS, las cuales requieren un alto grado de automatización y digitalización. Uno de los objetivos principales de los sistemas UTM es automatizar los trámites con los operadores para evitar saturar a la autoridad aérea con una gran cantidad de peticiones.

Por lo tanto, UTM establece distintos niveles automatización. Por un lado la correspondiente a las gestiones entre pilotos y autoridad aérea, y por otro lado, la automatización del plan de vuelo llevado a cabo en las operaciones, la cual veremos en la Sección 3.3.

3.2.1. SESAR

SESAR (Single European Sky ATM Research) fue un programa de I+D, iniciado en 2004, con el cometido de modernizar y homogeneizar el ATM europeo. En 2007, la Unión Europea y Eurocontrol fundan SESAR JU (Joint Undertaking), una gran colaboración público-privada entre los principales agentes en el sector aeronáutico internacional, entre los que se encuentran organismos administrativos del espacio aéreo y grandes fabricantes del sector de la aviación.

3.2.2. U-Space

U-Space es el proyecto para establecer un sistema UTM en el espacio aéreo europeo. Para ello, la CE ha desarrollado un marco regulatorio, plasmado en gran medida en el Reglamento [8]. El plan de implementación de U-Space fué desarrollado por SESAR JU, con el objetivo de hacer que U-space sea operativamente posible. En España, el Ministerio de Transportes, Movilidad y Agenda Urbana ha fijado el Plan de Acción Nacional para el Despliegue del U-space (PANDU), con el fin de implantar los servicios U-Space.

U-Space va a proporcionar los servicios para la navegación segura de UAS, además de una interfaz que integre las aeronaves tripuladas, los proveedores de servicios aéreos y las autoridades aéreas de cada país.

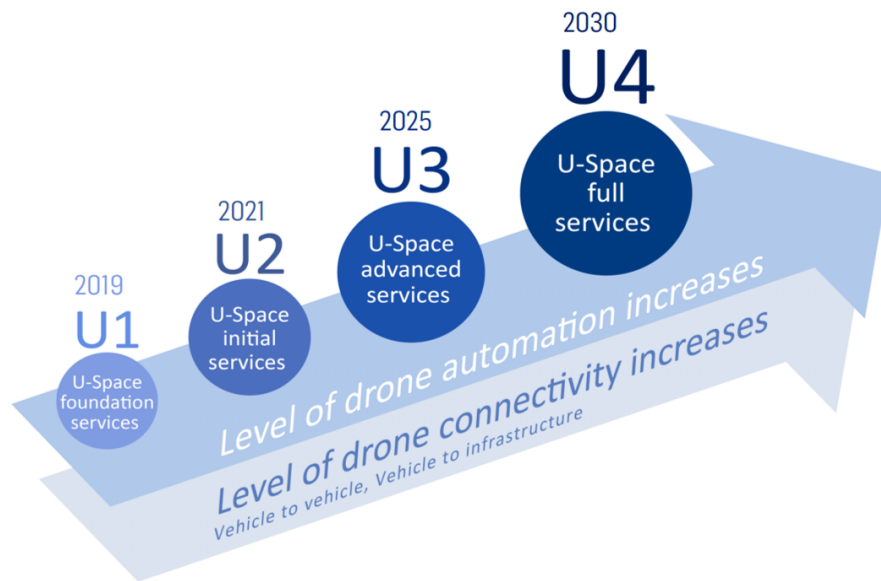


Figura 8: Fases U-Space

El plan de la SESAR JU propone la implementación de cuatro fases o grupos de servicios que podemos observar en la Figura 8.

- Fase U1 Servicios Básicos

La primera fase de implementación de U-Space incluye servicios de identificación electrónica de pilotos y aeronaves, así como servicios de información, sobre zonas áreas restringidas (geofencing) de forma previa a la operación. El objetivo principal de la fase U1 es facilitar cualquier clase de operativa, desde la más sencilla en zonas con densidad poblacional baja, hasta las más complejas en entornos urbanos de alta densidad.

- Fase U2 Servicios Iniciales

Esta segunda fase comprende a un conjunto de servicios que proporcionen seguridad a las operaciones con UAS, entre los que se incluye:

- Geofencing táctico que proporcione a los operadores de UAS la información actualizada sobre las zonas áreas restringidas.
- Rastreo de UAS e información del tráfico aéreo que pueda estar presente en

la zona de operación.

- Gestión digital de los planes de vuelo.
- Informaciones meteorológicas para apoyar al operador de UAS a la toma de decisiones operacionales.
- Gestión de situaciones de emergencias.
- Sistema de solicitud y aprobación de permisos, que garantice la exclusividad del espacio aéreo empleado en las operaciones UAS autorizadas.

La fase U2 también establece un primer nivel de interfaz y conexión con los servicios ATM y como consecuencia del incremento de la digitalización de procesos, permitirá un mayor número de operaciones que la fase U1.

■ Fase U3 Servicios Avanzados

La tercera fase dará soporte a operaciones complejas dentro del sistema U-space, mediante el desarrollo de los siguientes servicios:

- Geofencing dinámico, con los que detectar nuevas zonas áreas restringidas en pleno vuelo
- Interfaz colaborativa con el ATC para mantener una comunicación activa con los gestores de tránsito aéreo.
- Mitigación estratégica de aeronaves en vuelo mediante sistemas “detect and avoid” (DAA).

Estas nuevas tecnologías cooperativas permitirán un aumento significativo de operaciones en todos los entornos, además de un mejor soporte para la interacción con el ATC.

■ Fase U4 Servicios Completos

La fase final de implementación supone la integración completa de los protocolos de U-Space, los cuales proporcionarán apoyo completo a los UAS, en un nivel

muy elevado de automatización, conectividad y digitalización. La fase U4 incluye algunos servicios adicionales como la posibilidad de interconexión entre todas las aeronaves que circulen por el espacio aéreo, integrando de una manera efectiva a los operadores de UAS con el ATM.

A fecha del trabajo (Junio 2023) nos encontramos en la fase U2, en la que se han lanzado proyectos con el objetivo de probar y demostrar los servicios U-Space. Uno de estos proyectos es U-Elcome, en el que participa más de 50 entidades de España, Francia, Polonia e Italia, que desarrollarán operaciones comerciales con UAS, con el fin de introducir en el mercado los servicios U-Space de las fases U1 y U2.

3.3. Planificación de vuelos autónomos

La planificación de vuelos autónomos con drones consta de una preparación de la operativa previa al vuelo. El piloto define de la ruta que va a seguir el dron, mediante un conjunto de puntos o waypoints que marcan la trayectoria del vuelo. También se definen otros parámetros referentes al vuelo, tales como la altitud, velocidad, dirección de la cámara, etc. Durante el vuelo, el piloto puede verificar de forma remota el estado del dron, incluso modificar los parámetros de la ruta. Es necesario que haya al menos un piloto en campo, por si debe tomar el control del dron por razones de seguridad.

Para llevar a cabo un vuelo de forma autónoma, los drones cuentan con un autopiloto, el cual gestiona los datos de los sensores para dar órdenes al dron en función de los comandos recibidos, lo que permite al UAS seguir la ruta planificada, compensando los efectos del viento y otras perturbaciones de vuelo. En caso de pérdida de GPS el autopiloto pasa a comandar un hovering (vuelo estacionario sobre un punto) y en un tiempo preestablecido si no se recupera esta señal, se procederá al aterrizaje vertical de la aeronave.

Existen distintas herramientas para planificar el vuelo de un dron, entre las cuales nos hemos decantado por *Mission Planner*.

Para planificar vuelos autónomos con drones es necesario estar habilitado como piloto profesional de UAS en escenarios STS-ES-01 y STS-ES-02, ya que la mayoría de este tipo de vuelos se realizan en categoría específica.

La planificación de vuelos autónomos es necesaria en áreas que requieren planificar misiones con datos precisos, tales como la topografía, agricultura de precisión, seguridad o emergencias, entre otras. Además, resulta de especial interés con la implementación de los servicios de U-Space, ya que estos son prestados a operaciones con alto grado de automatización.

3.4. Partes de un plan de vuelo

El plan de vuelo no se queda en la definición de la ruta que va a seguir el dron. El piloto encargado de la operación deberá realizar una serie de procedimientos previos al vuelo, con el fin de comprobar que no existan elementos que puedan afectar a la seguridad de la operación.

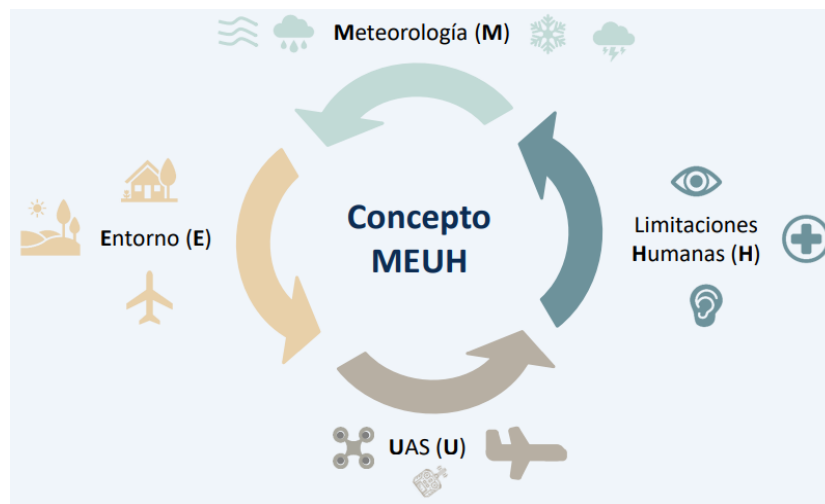


Figura 9: Partes Plan de Vuelo

Una de las metodologías empleadas para realizar el plan de vuelo se base en el “Concepto MEUH”, que podemos observar en la Figura 9.

3.4.1. Meteorología

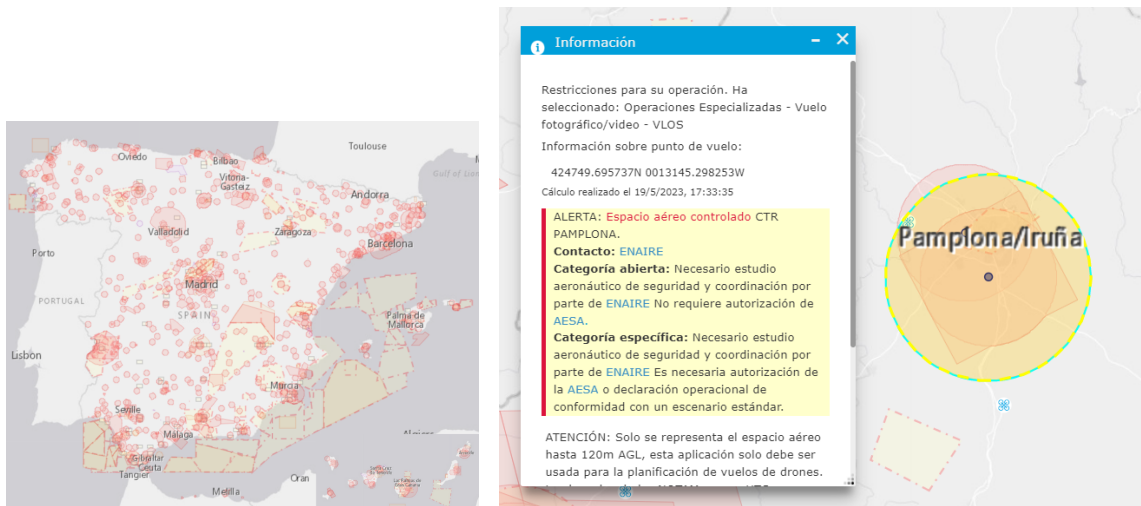
El piloto debe consultar las condiciones meteorológicas de forma previa a la operación y durante la misma. Estas deben ser compatibles con los límites definidas por el fabricante del dron. El piloto deberá tener en cuenta los efectos que producen distintos factores meteorológicos en el rendimiento de la aeronave y que puedan poner el riesgo el desarrollo de la operación. El viento puede afectar a la autonomía de la aeronave, ya que si es intenso esta necesitará una mayor energía para mantener la estabilidad. La temperatura también puede afectar al rendimiento de las baterías. El nivel de luz o la niebla afectan a la capacidad del piloto para visualizar la aeronave.

Si durante la operación el piloto observa un cambio en las condiciones meteorológicas que pueda afectar a la seguridad y rendimiento de la aeronave, cancelará el vuelo.

3.4.2. Entorno

El piloto debe evaluar el entorno donde va a realizar los vuelos antes de iniciar la operación, verificando que la operación se ajusta a la normativa y cumple con las posibles restricciones impuestas en el área de operación. AESA ha elaborado la Guía [9] para definir las restricciones de vuelo de UAS. Estas restricciones se unen a las limitaciones de vuelo definidas en Subsección 2.3.2, pero en este caso dependen zona geográfica dónde se desarrolle la operación, en la que se puede restringir o excluir operaciones con UAS por motivos de seguridad pública, protección de infraestructuras y otros usuarios del espacio aéreo, protección de la privacidad y del medio ambiente.

ENAIRE Drones



(a) Restricciones de vuelo

(b) Restricción Aeródromo Pamplona

Figura 10: Restricciones de vuelo ENAIRE

En la aplicación web “ENAIRE Drones” [10], se pueden consultar las áreas en las que existen restricciones al vuelo de drones, las cuales podemos observar en la Figura 10(a). ENAIRE publica restricciones formadas por puntos, como la capa vectorial de Aeródromos, con la localización de aeropuertos y helipuertos, y formadas por polígonos, como los CTR (Espacio Aéreo de Control), como el CTR del Aeródromo Pamplona que vemos en la Figura 10(b), en la que se especifica las autorizaciones necesarias por cada categoría operacional.

Algunas restricciones tienen carácter permanente o son cambios temporales de larga duración, como por ejemplo las alertas y avisos de la AIP (Publicación de Información Aeronáutica), que incluyen los CTR de aeródromos, o los Espacios Naturales Protegidos y Zonas de especial protección para aves (ENP/ZEPA). También se publica una capa con carácter temporal, los NOTAMs, avisos temporales de posibles peligros a lo largo de una ruta de vuelo.

Una vez comprobadas las restricciones en la zona de operación, el piloto debe cercio-

rarse de que no existan obstáculos que puedan afectar a la seguridad de la operación o interferir con la ruta planeada, tales como edificios, vehículos, vías públicas, montañas, árboles, antenas, líneas de tensión, etc.

3.4.3. UAS

Según el Reglamento [5], el operador de UAS tiene la responsabilidad de mantener el UAS en condiciones apropiadas para la operación que vaya a llevar a cabo. El mantenimiento dependerá de la categoría en la que se esté realizando la operación, siendo más estricto conforme aumenta el riesgo de la misma. Para operaciones en categoría abierta, el operador puede realizar el mantenimiento mediante sus propios procedimientos, pero basándose en el manual del fabricante. En cambio, para operaciones en categoría específica, debe ser realizado por personal debidamente formado que siga las instrucciones de mantenimiento. Los requisitos de mantenimiento para la categoría certificada se encuentran en proceso de definición por parte de EASA.

3.4.4. Limitaciones Humanas

Antes de iniciar una operación, el piloto debe verificar que se encuentra en condiciones óptimas para llevar a cabo la operación. Para ello, puede emplear la checklist IMSAFE:

- Enfermedad (*Illness*)
- Medicación (*Medication*)
- Estrés (*Stress*)
- Alcohol (*Alcohol*)
- Fatiga (*Fatigue*)
- Emociones (*Emotions*)

Servicio de información meteorológica

En este capítulo vamos a detallar el desarrollo llevado a cabo para ayudar al piloto a mantenerse informado de las condiciones meteorológicas, tal y como vimos en la Subsección 3.4.1. Además de información meteorológica, vemos interesante recoger información sobre el aforo de los cauces, debido a que las operaciones dentro del proyecto EMERAL son realizadas en las inmediaciones de los ríos. En concreto, hemos implementado un servicio que recoge datos de sensores instalados en aforadores y estaciones meteorológicas ubicadas en Navarra.

4.1. Conjunto de datos

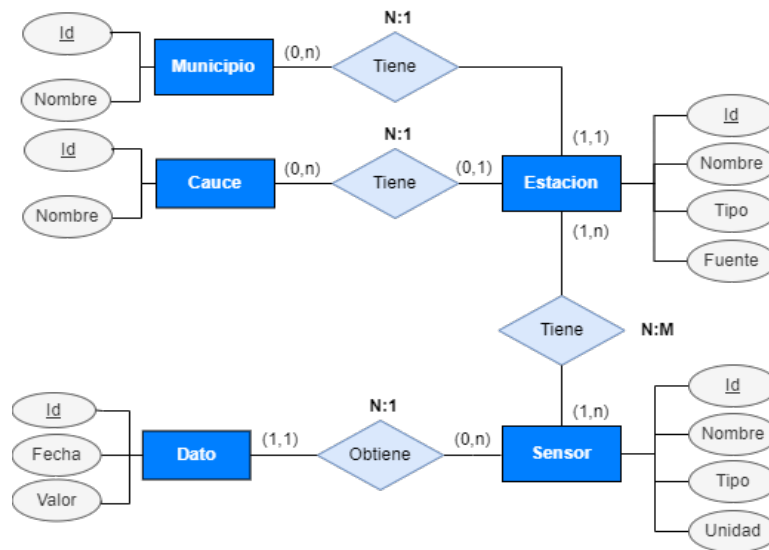


Figura 11: E/R datos meteorológicos y de aforo

En la Figura 11 vemos el diagrama Entidad / Relación del conjunto de datos obtenidos en el servicio.

Entidades

Todas las entidades presentan un identificador único por cada uno de sus registros. Tanto de los municipios como de los cauces, solamente recogemos el nombre. En el caso de las estaciones, además las diferenciamos por su tipo (Meteorológica o De aforo) y por las fuentes, estas últimas las veremos en la Sección 4.2. Los sensores también están agrupados por tipos (Nivel, Caudal, Precipitación, Viento, Temperatura y Humedad) y guardamos la unidad de medida de los datos generados por el sensor. Es importante aclarar que la entidad Sensor no hace referencia al dispositivo físico, sino al conjunto de valores que obtiene. Por ejemplo, en el caso de los sensores de tipo “Viento”, tenemos unos que miden la velocidad del viento y otros que miden la dirección del viento. Finalmente, obtenemos los datos meteorológicos y de aforo mediante un valor y la fecha de recogida.

Relaciones

- Municipio - Estación: En un municipio puede haber algunas estaciones o ninguna, y una estación siempre estará ubicada en un único municipio.
- Cauce - Estación: En un cauce puede haber algunas estaciones o ninguna, y una estación puede estar o no ubicada en un cauce, dependiendo de su tipo. Los aforadores siempre estarán ubicados en un cauce, pero las estaciones meteorológicas no.
- Estación - Sensor: Las estaciones tiene instalados al menos un sensor, y un sensor puede estar instalado en una o muchas estaciones.
- Sensor - Dato: Un sensor genera datos, aunque también nos hemos encontrado con sensores que no generan ningún dato, probablemente por mal funcionamiento. Un dato es generado por un único sensor.

Una vez definido los campos, entidades y las relaciones, vamos a ver de dónde hemos obtenido los datos y cómo los hemos procesado y expuesto.

4.2. Fuentes de datos

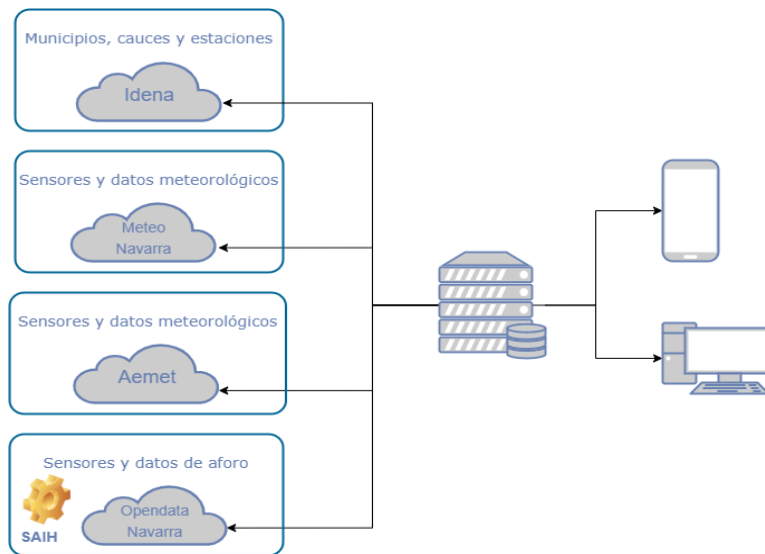


Figura 12: Fuentes de información

En la Figura 12 vemos las fuentes de la información obtenida.

4.2.1. IDENA

La Infraestructura de Datos Espaciales de Navarra (IDENA) es una red de recursos de información referidos al territorio de la Comunidad Foral de Navarra. Expone información geográfica de todo tipo: catastral, municipal, hidrográfica, meteorológica, cartográfica, etc.

Tal y como vimos en [11], IDENA expone un Servicio de Objetos Espaciales o *Web Feature Service* (WFS), con el que obtener capas vectoriales, formadas tanto por propiedades geométricas (puntos, líneas, polígonos), como propiedades descriptivas (nombre, descripción, área, etc.). Estas capas se pueden descargar en formato Shapefile, que es el más extendido y popular para el almacenamiento de datos vectoriales.

En concreto, hemos obtenido las siguientes capas:

- Municipios

The screenshot shows a Postman REST client interface. The top bar indicates a POST request to the URL: `https://idena.navarra.es/ogc/wfs?service=WFS&version=2.0.0&typename=IDENA:CATAST_Pol_Municipio&srsname=EPSG:4326&request=GetFeature&outputFormat=json`. The 'Query Params' section contains a table with the following data:

Key	Value	Description
<input checked="" type="checkbox"/> service	WFS	
<input checked="" type="checkbox"/> version	2.0.0	
<input checked="" type="checkbox"/> typename	IDENA:CATAST_Pol_Municipio	
<input checked="" type="checkbox"/> srsname	EPSG:4326	
<input checked="" type="checkbox"/> request	GetFeature	
<input checked="" type="checkbox"/> outputFormat	json	

The 'Body' section shows the JSON response in 'Pretty' format:

```
1 {
2   "type": "FeatureCollection",
3   "totalFeatures": 342,
4   "features": [
5     {
6       "type": "Feature",
7       "id": "CATAST_Pol_Municipio.1",
8       "geometry": {
9         "type": "MultiPolygon",
10        "coordinates": [
11          [
12            [
13              [
14                -0.99830965,
15                42.95653836
```

Figura 13: Descarga de la capa Municipios

La capa está formada por 342 polígonos definen los límites administrativos de cada municipio. Podemos ver su descarga mediante la herramienta *Postman* en la Figura 13.

- Cauces

La capa está formada por 3418 líneas que definen la localización de las corrientes naturales de agua.

- Aforadores o estaciones de aforo

La capa está formada por 59 puntos que definen la ubicación de las estaciones de aforo.

- Estaciones meteorológicas

Hemos recogido información sobre estaciones meteorológicas a partir de dos capas de puntos que definen la ubicación de las estaciones meteorológicas. Una capa corresponde a las estaciones automáticas, con un total de 55 estaciones, y otra

capa con las estaciones manuales, que son 80. Una misma estación puede ser automática y manual, lo cual comprobaremos a partir de las localizaciones de las estaciones.

4.2.2. AEMET

La Agencia Estatal de Meteorología (AEMET), adscrita al Ministerio para la Transición Ecológica y el Reto Demográfico, expone información meteorológicos a nivel nacional, a través de una API REST. En la web de AEMET, hay una sección para desarrolladores con ejemplos de programas cliente en diferentes lenguajes de programación. También podemos ver y probar todas las peticiones de la API desde [12].

En concreto, hemos obtenido los datos de las estaciones mediante la llamada `/api/observacion/convencional/datos/estacion/{idema}`, siendo *idema* el indicativo climatológico de la EMA (Estación meteorológica automática), que es el código que identifica a cada estación. Las estaciones de Navarra junto a su *idema* las hemos obtenido del archivo XML <https://opendata.aemet.es/centrodedescargas/xml/udat/udat31.xml>.

4.2.3. Meteo Navarra

Meteo Navarra es una plataforma web que expone datos meteorológicos de Navarra en tiempo real. Esta fuente recoge datos registrados en las estaciones meteorológicas del Gobierno de Navarra, AEMET, SAIH Ebro, INTIA (sociedad pública, adscrita al Departamento de Desarrollo Rural y Medio Ambiente del Gobierno de Navarra) y MAPAMA (Ministerio de Agricultura, Pesca y Alimentación).

Meteo Navarra permite la descarga de información meteorológica en los formatos *XLS*, *CSV* y *ODS*. En nuestro caso hemos descargado ficheros *CSV* que posteriormente hemos procesado.

Pamplona (ETSIA) UPNA

ESTACIÓN AUTOMÁTICA

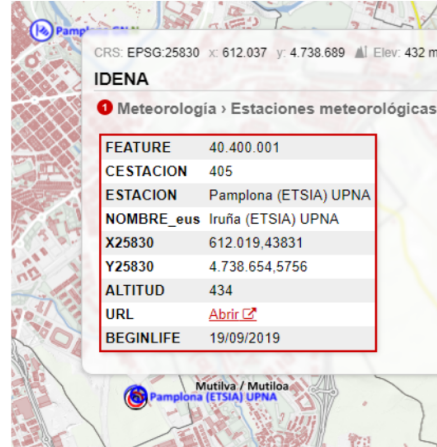
Altitud: 433 m

Propiedad de Universidad Pública de Navarra



X:612019 Y:4738655 (*)

Instalación: 01/04/2004



1. Parámetros

Parámetros 10 minutos

- Temperatura
- Humedad relativa
- Radiación global
- Insolación
- Velocidad viento 10 m
- Dirección viento 10 m
- Velocidad racha máxima 10 m

Parámetros Diarios

- Temperatura media
- Temperatura máxima
- Temperatura mínima
- Humedad relativa med.
- Humedad relativa máx.
- Humedad relativa mín.
- Precipitación acumulada

Figura 14: Estación Meteorológica UPNA

En concreto, hemos recogido datos de las estaciones meteorológicas obtenidas de IDE-NA. Un ejemplo de ello son los datos de la estación situada en la Universidad Pública de Navarra, que tal y como vemos en la Figura 14, Meteo Navarra nos permite descargar una serie de parámetros diez-minutales y diarios. También podemos observar que las coordenadas de la estación coinciden en la web de Meteo Navarra y en el visor de IDENA.

Meteo Navarra permite filtrar los datos de los parámetros/sensores mediante fecha inicial y fecha final. Por lo tanto, haremos uso de dicho filtro para obtener los datos más recientes y ofrecer al piloto la posibilidad de obtener datos de días concretos.

4.2.4. OpenData Navarra

OpenData Navarra es una iniciativa del Gobierno de Navarra para exponer datos públicos sin restricciones técnicas ni legales, con el fin de que particulares y empresas puedan consultados, distribuirlos y crear herramientas y servicios. Presenta un catálogo de datos muy amplio, relacionados con el medio ambiente, el sector público, urbanismo, transporte, economía, salud, sociedad, ciencia, entre otros sectores. Entre todos los conjuntos de datos que ofrece el OpenData Navarra, obtenemos datos del Sistema Automático de Información Hidrológica (SAIH).

El SAIH ofrece valores diez-minutales sobre el nivel y el caudal de algunos ríos de Navarra. Esta información se expone a través de una API y se puede obtener mediante la llamada https://datosabiertos.navarra.es/es/api/3/action/datastore_search?resource_id=c736f9a3-3efc-4b89-9d31-ef859607bc27.

4.3. Procesamiento de datos

Una vez conocidas todas las fuentes y las formas en las que estas exponen los datos, vamos a ver cómo gestionar el conjunto total de datos descargados, con el objetivo de unificar en un formato toda la información obtenida.

Para ello, hemos empleado el lenguaje de programación *Python* y en concreto las librerías *Pandas* y *Geopandas*, que permite volcar la información en unas estructuras de datos llamadas *DataFrame* y *GeoDataFrame*, respectivamente. Estas librerías están optimizadas para el manejo de grandes cantidades de datos, por lo que resultan eficientes en cuanto al tiempo de su procesamiento. Disponen los datos en forma de tabla, en la que cada fila corresponde a un vector que guarda las propiedades de los objetos que almacena. En los *GeoDataFrames*, además de almacenar las propiedades descriptivas, también almacena las propiedades geométricas.

Vemos un ejemplo con la descarga de información sobre los municipios de Navarra del WFS de IDENA:

```
[1]: import requests
import geopandas as gpd
url = "https://idena.navarra.es/ogc/wfs"
parameters = {
    'service': 'WFS',
    'version': '2.0.0',
    'typename': 'IDENA:CATAST_Pol_Municipio',
    'srsname': 'EPSG:4326',
    'request': 'getFeature',
    'outputFormat': 'json'
}
response = requests.post(url, data=parameters)
json = response.json()
municipiosGDF = gpd.GeoDataFrame.from_features(json["features"])
```

```
[1]: geometry FEATURE CMUNICIPIO
↳MUNICIPIO \
190 MULTIPOLYGON (((-1.84177 42.84037, -1.84189 42... 200020 194
↳Valle de Olló
282 MULTIPOLYGON (((-1.63079 42.77244, -1.63063 42... 200020 88
↳Noáin (Valle de Elorz)
297 MULTIPOLYGON (((-2.03778 42.80587, -2.03807 42... 200020 260
↳Valle de Yerri
337 MULTIPOLYGON (((-1.57393 42.83506, -1.57391 42... 200020 86
↳Valle de Egüés
```

Una vez que hemos importado los datos, podemos filtrarlos para quedarnos con aquellos que nos interesan. En el caso de los municipios, hemos descartado las facerías, que según el Fuero Nuevo de Navarra de 1973, se definen como una servidumbre recíproca entre varias fincas de propiedad colectiva o privada. En cuanto a los cauces nos hemos quedado con los ríos principales, para ello hemos filtrado el nombre de los datos recibidos por

aquellos que contienen las palabras “RIO” o “IBAIA”.

En las siguientes líneas de código vemos cómo se realizan ambos filtrados:

```
[2]: municipiosGDF.loc[~municipiosGDF.Nombre.str.contains("Facería")]
caucesGDF[caucesGDF.Nombre.str.contains('RIO |IBAIA ', regex=True, na=False)]
```



Figura 15: Capas Municipios Cauces y Estaciones

En la Figura 15 vemos las capas vectoriales obtenidas de IDENA después de realizar el filtrado. Hemos empleado el visor de capas de la herramienta *QGIS* para visualizar las tres capas resultantes: la capa de municipios representada mediante polígonos amarillos, la capa de cauces con líneas azules y la capa de estaciones con puntos rojos.

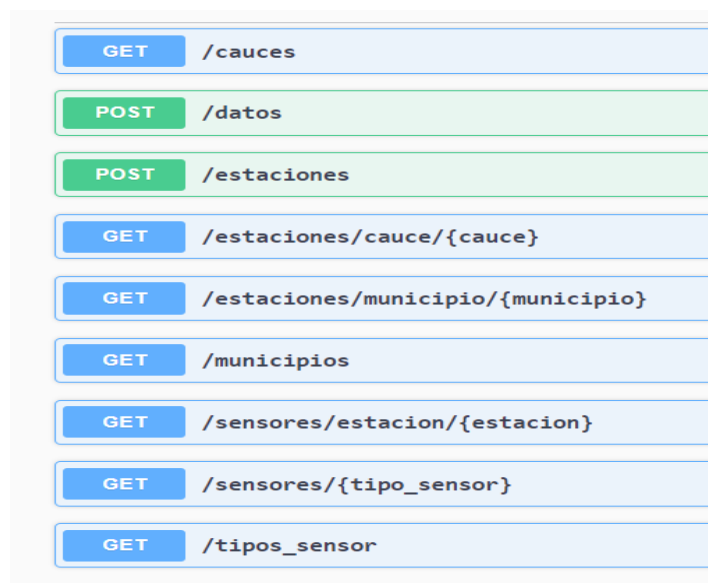
Para establecer las relaciones Municipio - Estación y Cauce - Estación, hemos realizado operaciones con las geometrías de cada objeto. En concreto, hemos determinado las estaciones de un municipio por los puntos (estaciones) que contiene, y hemos determinado las estaciones de un cauce, calculando la distancia más cercana de cada aforador a las líneas (cauces). El código empleado es el siguiente:

```
[3]: [municipiosGDF.contains(geo) for _ , geo in estacionesDGF.geometry.items()]
      [aforadoresGDF.distance(geo).idxmin() for _ , geo in estacionesDGF["geometry"].
      ↪items()]
```

En cuanto a los sensores, hemos consultado en cada fuente los parámetros que exponen por cada estación. Una vez establecidas todas las relaciones, simplemente es cuestión de realizar consultas de los datos desde nuestro servicio y que este consulte a las fuentes mencionadas la misma información. De esta forma permitimos al piloto consultar datos meteorológicos del territorio navarro desde un único punto.

4.4. Despliegue

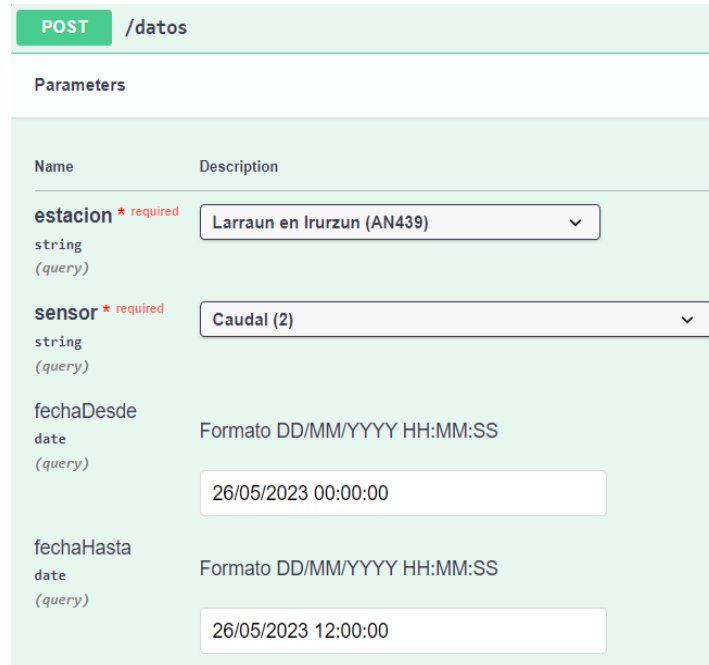
El procesamiento para obtener información meteorológica y de aforo se ha materializado en forma de API, empleando el framework *Flask* de *Python*, el cual permite implementar un servicio web de forma sencilla y rápida. Hemos empleado *Swagger* para documentar y utilizar el servicio, que es la misma herramienta que utiliza AEMET en [12]. *Swagger* nos permite listar los métodos que expone el servicio y hacer peticiones por cada uno de ellos.



GET	/cauces
POST	/datos
POST	/estaciones
GET	/estaciones/cauce/{cauce}
GET	/estaciones/municipio/{municipio}
GET	/municipios
GET	/sensores/estacion/{estacion}
GET	/sensores/{tipo_sensor}
GET	/tipos_sensor

Figura 16: Métodos Servicio Meteorológico e Hidrológico

El servicio meteorológico e hidrológico en el que el piloto puede consultar información, está publicado en [13] y en la Figura 16 vemos el conjunto de métodos que se exponen.



The image shows a REST client interface for a POST request to the endpoint `/datos`. The parameters are defined as follows:

Name	Description
estacion * required string (query)	Larraun en Irurzun (AN439)
sensor * required string (query)	Caudal (2)
fechaDesde date (query)	Formato DD/MM/YYYY HH:MM:SS 26/05/2023 00:00:00
fechaHasta date (query)	Formato DD/MM/YYYY HH:MM:SS 26/05/2023 12:00:00

Figura 17: Métodos Servicio Meteorológico e Hidrológico

El más importante es el método POST de `/datos`, con el que podemos obtener la información de un sensor correspondiente a una estación en un rango de fechas, tal y como vemos en la Figura 17.

Cabe destacar que las fuentes de información pueden modificar el formato de los datos, el nombre de los campos, los endpoints de los métodos, etc. Por lo tanto, es recomendable comprobar el funcionamiento del nuestro servicio cada poco tiempo, para que en caso de modificaciones, lo podamos adaptar de la forma más rápida posible. Incluso, podemos implementar un sistema de peticiones automáticas que nos envíe una alerta cada vez que se produce un error.

Identificación de restricciones de vuelo

En este capítulo vamos a detallar el desarrollo llevado cabo para identificar de manera automática las restricciones que incumple el vuelo del dron. Comenzaremos describiendo las zonas total o parcialmente restringidas al vuelo de drones y dónde se pueden consultar. También veremos la forma de descargar y gestionar esta información y cómo presentársela al piloto para agilizar el proceso de planificación del vuelo.

5.1. Restricciones de vuelo de drones

Como vimos en la Subsección 3.4.2, el piloto debe verificar que la operación cumple con la Guía [9], en cuanto a las restricciones de vuelo de UAS, las cuales puede consultar en la aplicación [10]. Esta aplicación permite visualizar en el mapa el conjunto de restricciones de vuelo, dibujar geometrías e importarlas desde un fichero de coordenadas. La forma que tienen los pilotos de comprobar si la operación no incumple ninguna restricción, es dibujando la ruta en el mapa o cargando el fichero con las coordenadas de la ruta del vuelo. Esta manera de proceder resulta lenta, sobretodo en planes de vuelo en situaciones de emergencia, los cuales requieren de una mayor agilidad. Por ello, vamos a implementar un sistema automático que identifique las restricciones.

5.1.1. Capas de restricciones

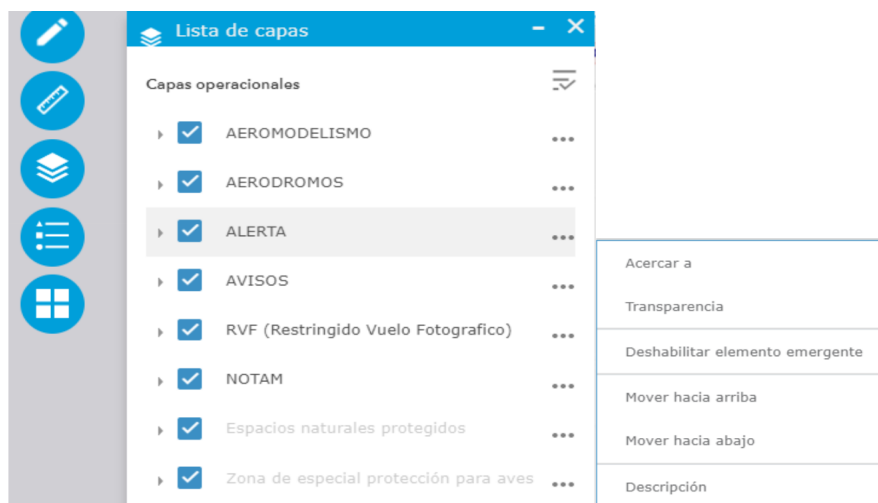


Figura 18: Listado capas ENAIRE

En la Figura 18 podemos ver el listado de las capas vectoriales que muestra el mapa interactivo de la aplicación [10].

- **AEROMODELISMO**

El aeromodelismo es un deporte en el que se construye y vuela aeromodelos, réplicas a escala de aviones existentes o diseños de prueba para futuros aviones reales. La capa está formada por puntos, por ejemplo, contiene la localización del campo de vuelo de la Asociación Club Navarra de Aeromodelismo cerca de Lizasoain.

- **AERODROMOS**

Según el Ministerio de Transportes, Movilidad y Agenda Urbana, un Aeródromo es el “Área definida de tierra o de agua (que incluye todas sus edificaciones, instalaciones y equipos) destinada total o parcialmente a la llegada, salida y movimiento en superficie de aeronaves de ala fija”. Esta capa también está formada por punto y, aunque la capa tenga el nombre de AERODROMOS, incluye tanto aeródromos, como aeropuertos y helipuertos.

- ALERTA

La capa contiene polígonos que definen áreas de seguridad o CTRs (Espacio Aéreo de Control), debido a distintos motivos que afectan al vuelo de aeronaves, tales como, las inmediaciones de aeropuertos o hospitales.

- AVISOS

La capa contiene polígonos que definen zonas en la que pueden desplegarse actividades peligrosas para el vuelo de aeronaves, como por ejemplo, el campo de entrenamiento de tiro aéreo situado en las Bardenas Reales.

- RVF (Restringido Vuelo Fotográfico)

La capa está formada por polígonos que definen las inmediaciones de lugares considerados estratégicos por el Ministerio de Defensa, como instalaciones militares o gubernamentales. Para realizar fotos o vídeos en estas zonas, se requiere solicitar permiso al Estado Mayor del Aire (EMA).

- NOTAM (NOTice To AirMen)

Como ya hemos comentado anteriormente, los NOTAM son avisos temporales de posibles peligros para la seguridad aérea. La capa está formada por polígonos similares a los de la capa AVISOS, pero de carácter temporal.

- ENP (Espacios Naturales Protegidos)

La capa contiene polígonos que definen zonas naturales protegidas, como las que determina Red Natura 2000, una red ecológica europea de áreas de conservación de la biodiversidad. También incluye parques nacionales o áreas marinas protegidas entre otras zonas.

- ZEPA (Zonas de especial protección para aves)

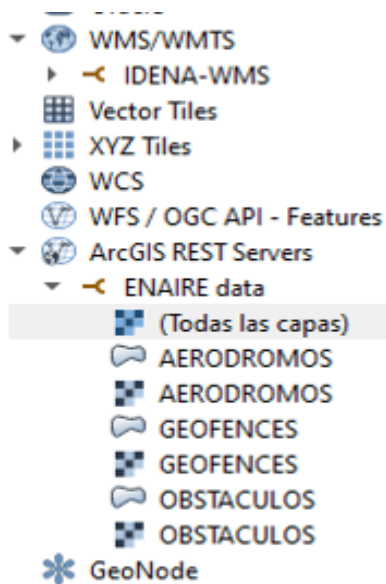
La capa contiene polígonos con las “zonas naturales de singular relevancia para la conservación de la avifauna amenazada de extinción” (Comisión Europea).

Una vez conocidas las distintas restricciones de vuelo de drones, nos planteamos el objetivo de obtener y gestionar esta información de la manera más eficiente posible.

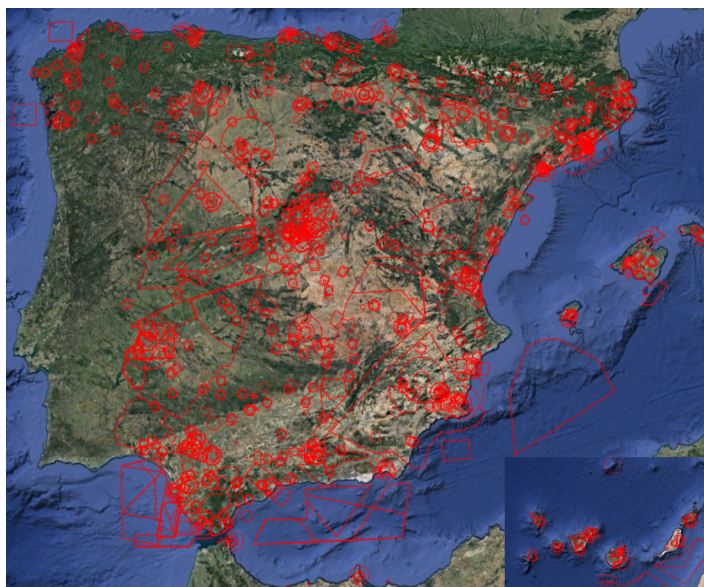
5.1.2. Fuente de información de ENAIRE

Si accedemos a la descripción de cada capa, la aplicación nos redirige a un servicio “ArcGIS REST Service”, el cuál almacena dicha información. ENAIRE no dispone de una API que exponga los datos que se muestran en el mapa de [10], sino que los consume de un servicio ArcGIS. Por ejemplo las cinco primeras capas (AEROMODELISMO, AERODROMOS, ALERTA, AVISOS y RVF) de la Figura 18 las obtiene de [14].

En [11], vimos las diferencias entre las dos principales aplicaciones GIS (Geographical Information System): *ArcGIS* y *QGIS*. *ArcGIS* es de pago, comercializada por ESRI (Environmental Systems Research Institute), e incluye un conjunto de productos software, entre los que se encuentran aplicaciones de escritorio (*ArcGIS Desktop*) o servicios de publicación de datos geográficos (*ArcGIS Server*), como el empleado por ENAIRE en [10]. Por su parte, *QGIS* es el líder de código abierto para escritorio, el cuál ya hemos utilizado para visualizar los datos de la Figura 15.



(a) Capas ENAIRE en QGIS



(b) Capas ENAIRE en Google Earth Pro

En la Figura 19(a) vemos las capas que expone el servicio ArcGIS empleado por ENAIRE, las cuales hemos importado en *QGIS* creando una conexión con ArcGIS REST

Server. Una vez cargadas las capas en *QGIS*, podemos visualizarlas y exportarlas en formato KML (Keyhole Markup Language). KML está basado en la sintaxis del formato XML y desarrollado para utilizarse en la herramienta *Google Earth*. La Figura 19(b) muestra las restricciones de vuelo en la herramienta *Google Earth Pro*, las cuales coinciden con las restricciones mostradas en la Figura 10(a).

5.1.3. Descarga de datos

Para la descarga de las capas publicadas en [14], hemos empleado *PyQGIS*, la librería de *QGIS* para ejecutar código *Python*.

```
[1]: from qgis.core import QgsVectorLayer, QgsVectorFileWriter
      # Get Vector Layer of ArcGis Web Feature Service
      url = "https://servais.enaire.es/insigniad/rest/services/NSF_APP/Drones_APP_V1/
↳MapServer/1"
      uri = f"crs='EPSG:4326' url='{url}'"
      layer_name = "ALERTAS"
      vlayer = QgsVectorLayer(uri, layer_name, "arcgisfeatureserver")
      # Check if layer is valid
      if vlayer.isValid():
          # Create layer shapefile
          path = 'C:\\\\ENAIRES\\{layer_name}'
          QgsVectorFileWriter.writeAsVectorFormat(vlayer, path, 'utf-8', vlayer.
↳crs(), 'ESRI Shapefile')
```

En el código anterior, descargamos la información correspondiente a las alertas, creando la clase *QgsVectorLayer*, el cuál podemos guardar en formato Shapefile para su posterior tratamiento. Debido al carácter temporal de los avisos NOTAMs, sobre posibles peligros a lo largo de una ruta de vuelo, es necesario descargar dicha capa cada poco tiempo. Para ello, hemos desarrollado la aplicación *app.py* con el código anterior adaptado a la capa NOTAM y hemos creado una tarea programada de Windows para que ejecute esta aplicación todos los días.

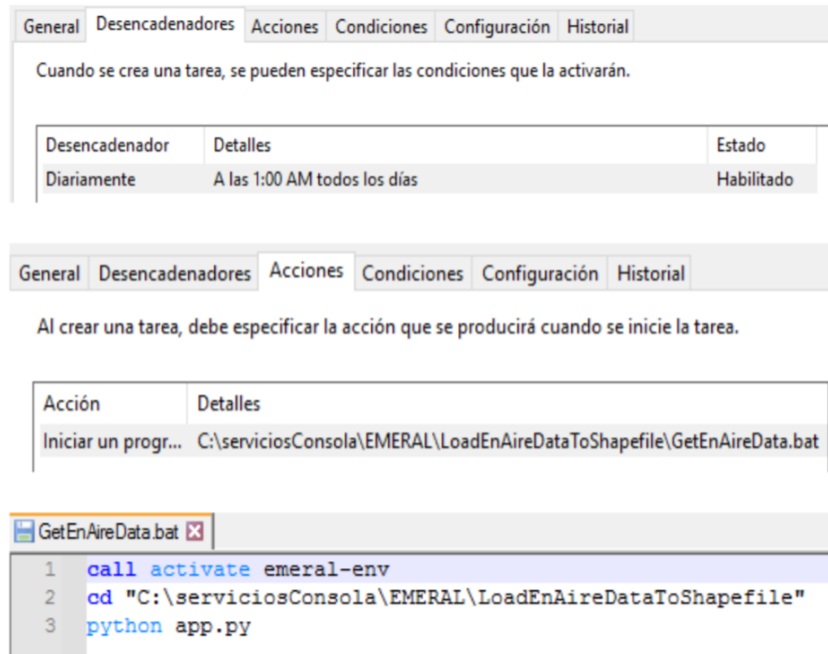


Figura 19: Tarea programada de Windows descarga datos ENAIRE

En la Figura 19 vemos la configuración de la tarea programada, la cuál ejecuta el archivo *GetEnAireData.bat* todos los días a las 1:00 AM. Podemos ver que lo primero que hace es activar el entorno de *Python* que contiene las librerías necesarias para ejecutar el programa *app.py*. Hemos empleado *Miniconda* como gestor de paquetes y de entornos. Posteriormente, nos situamos en el directorio dónde se encuentra el archivo *app.py* y lo ejecutamos.

5.1.4. Gestión de restricciones

Al igual que en Sección 4.3, hemos empleado la librería *Geopandas* para gestionar las información sobre las restricciones de vuelo de drones. Con *Geopandas* podemos importar los datos almacenados en los ficheros Shapefile a las estructuras de datos *GeoDataFrames*. Esta tecnología nos permite hacer operaciones con geometrías de forma rápida y sencilla.

```
[2]: import geopandas as gpd
      from shapely.geometry import LineString
      # Read Shapefile
      layer_name = "ALERTAS.shp"
      path = f"C:\\\\ENAIRES\\\\" + layer_name
      layerGDF = gpd.read_file(path.join(path, layer_name))
      # Input drone route as a LineString
      points = [(-1.344997, 42.723154), (-1.342160, 42.729421)]
      line = LineString([points])
      # Get ENAIRE restrictions that intersect with the route
      restrictionsGDF = gpd.GeoDataFrame(layerGDF[layerGDF.intersects(line)])
```

En el código anterior, importamos la capa vectorial correspondiente a las restricciones de vuelo ocasionadas por las alertas aéreas. Posteriormente, definimos una línea que une dos puntos, (-1.344997, 42.723154) y (-1.342160, 42.729421), que representa la ruta de vuelo. Finalmente, filtramos las restricciones que intersectan con la línea especificada.

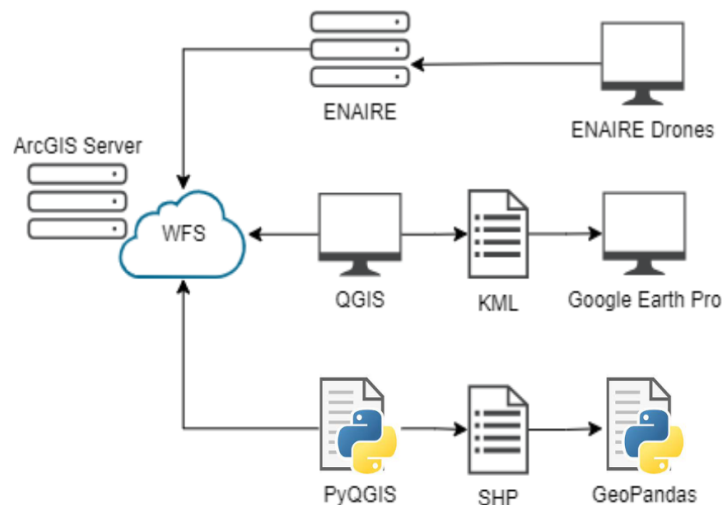


Figura 20: Servicio ArcGIS

En la Figura 20 vemos el esquema general de obtención de datos del WFS del Servicio ArcGIS. ENAIRE lo emplea para la aplicación [10], y nosotros, por un lado, empleamos *QGIS* y *Google Earth Pro* para validar los datos, y por otro, *PyQGIS* y *Geopandas* para la descarga y gestión de los datos, respectivamente.

5.2. Despliegue

En este apartado vamos a entrar más en detalle en el las soluciones llevadas a cabo por la empresa i3Code dentro del proyecto EMERAL, comentado en Subsección 1.1.1. En concreto, veremos cómo los pilotos pueden hacer uso de una aplicación web para consultar las restricciones de vuelo. En primer lugar, recibimos información referente al vuelo por parte de Tesicnor, a través de una API de desarrollo propio.

5.2.1. API Comunicación Tesicnor e i3Code

i3Code ha desarrollado una API para la comunicación de información con Tesicnor, con la que Tesicnor envía datos relacionados con el vuelo cada vez que solicita la composición de un plan de vuelo.

El formato de los mensajes se ha definido entre ambas partes, a medida que se han ido identificando nuevas necesidades tanto de Tesicnor, como de los pilotos que van a realizar el vuelo, aportados por los socios FuVeX, UPNA y NAITEC. A continuación, se especifica el esquema JSON que describe la estructura de los datos recibidos:

```
{
  uuid:      STRING ->Identificador único del vuelo.
  name:      STRING ->Nombre del vuelo.
  category:  CATEGORYETYPE ->Categoría del vuelo.
  priority:  INTEGER ->Prioridad del vuelo.
  home:      POINT ->Punto de despegue.
  points:    LIST<POINT>->Lista de waypoint de la ruta.
}

ENUM CATEGORYETYPE {
  MAINTENANCE, EMERGENCY
}
```

```

POINT {
    agl:          INTEGER ->Above ground level, altitud del dron sobre el suelo.
    poi:          BOOLEAN ->Point of interest punto especifico de interés potencial.
    description:  STRING ->Breve descripción del POI.
    restricted:   BOOLEAN ->Campo que indica si el suelo es urbano o no lo es.
    latitude:     DOUBLE ->Latitud del punto geográfico.
    longitude:    DOUBLE ->Latitud del punto geográfico.
}

```

En la ruta de prueba definida en Subsección 5.1.4, el json recibido sería el siguiente:

```

{"uuid": "6ba53aaa-bdac-4cd6-9ee9-a9c15512d0b4",
 "name": "Example Flight",
 "category": EMERGENCY,
 "priority": 1,
 "home": null,
 "points": [
   {
     "agl": 50,
     "poi": false,
     "description": null,
     "restricted": true,
     "latitude": 42.723154,
     "longitude": -1.344997
   },
   {
     "agl": 50,
     "poi": true,
     "description": "Example Point of Interest",
     "restricted": false,
     "latitude": 42.729421,
     "longitude": -1.342160
   }
 ]
}

```

Para recibir esta información se ha habilitado un método POST, llamado *Generate*, que devuelve un fichero con el plan de vuelo a partir de la información del vuelo.

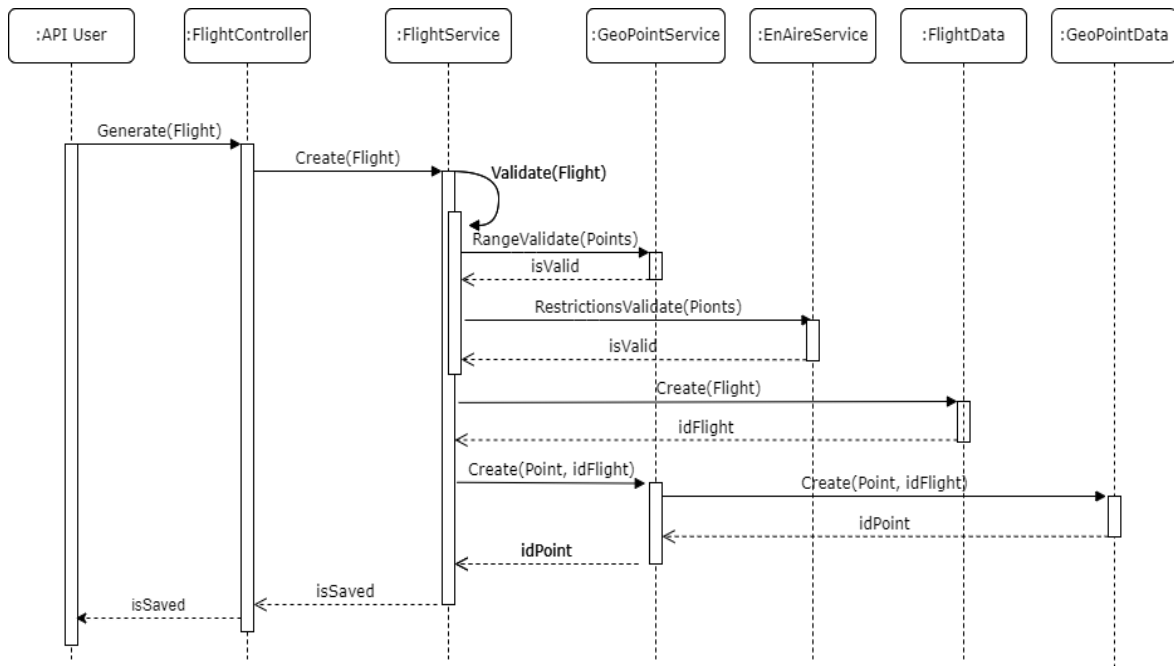


Figura 21: Diagrama secuencia método *Generate*

En la Figura 21 muestra el diagrama de secuencia del método *Generate*, en el que el objeto *Flight* contiene la información vista anteriormente. Vemos como cada uno de los waypoints que conforman el vuelo pasa por una fase de validación, en la que se comprueba que su latitud y longitud están contenidas en los rangos válidos. Los valores de la Longitud pueden ir de -180 a $+180^\circ$ y los de la Latitud de -90° a $+90^\circ$. También se comprueba que la ruta del vuelo no entra en ninguna zona restringida por ENAIRE, aunque como veremos a continuación, esto supone una razón suficiente para no guardar el vuelo en base de datos.

Finalmente, el método genera como salida un mensaje de confirmación si el vuelo se ha guardado correctamente, o un mensaje de error si algún punto no entra en el rango especificado o se ha producido algún fallo al guardar el vuelo.

5.2.2. Web EMERAL

Filters

UUID Name Category Priority dd/mm/aaaa

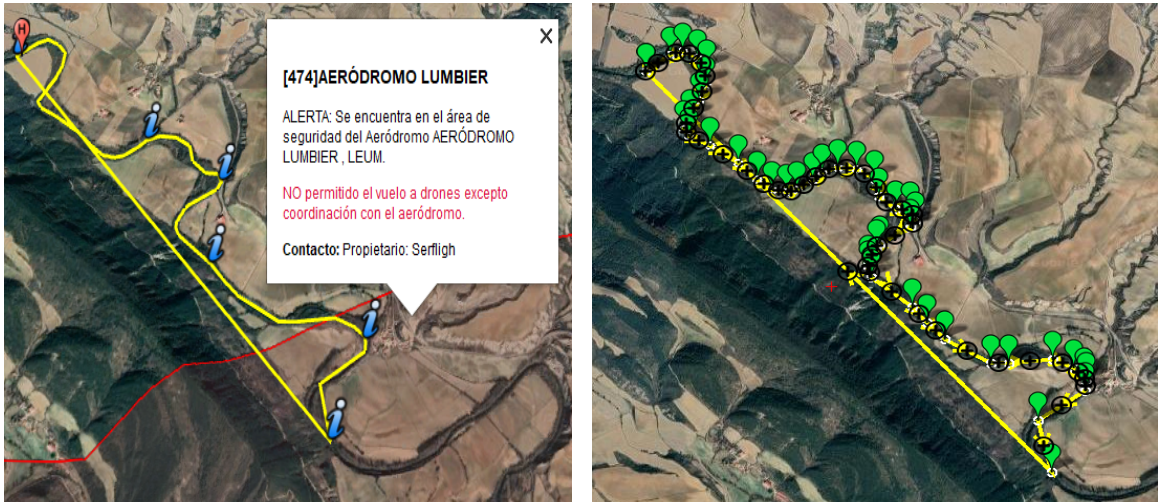
UUID	Name	Category	Priority	Create Date	Flight Request	Flight Plan	Flight Detail
c927867a-67f7-4b28-8324-22653eaf5660	Irati		1	21/04/2023 8:14:55	↓	↓	↓
59dc45f5-7cfd-4235-a0a9-f952fc2681d1	Example Flight	Category 1	1	03/02/2023 11:18:22	↓	↓	↓
c927867a-67f7-4b28-8324-22653eaf5660	Irati		1	01/02/2023 8:15:54	↓	↓	↓
13656abb-f569-4d31-9206-89c4cbcfe610	Example Flight	Category 1	1	26/01/2023 8:22:41	↓	↓	↓

Figura 22: Listado vuelos

i3Code también ha desarrollado una aplicación web para ver el histórico de vuelos que se han enviado a través de la API, tal y como vemos en la Figura 22, desde dónde los usuarios pueden descargar una serie de archivos:

- Flight Request: Archivo JSON recibido en la petición de generación de vuelo.
- Flight Plan: Archivo WAYPOINT con los comandos de vuelo, que se puede importar en la aplicación *Mission Planner* para que el dron realice el vuelo de forma autónoma.
- Flight Detail: Archivo KML con el detalle del plan de vuelo. Incluye la ruta del vuelo y en caso de que las hubiera, las restricciones de vuelo que se incumple.

Los registros de vuelos con el nombre “Example Flight” son peticiones realizadas por el desarrollador para testear el funcionamiento de la API. En cambio, el registro de vuelo más reciente, con fecha de 21 de Mayo de 2023, corresponde a una prueba de campo realizada en un tramo del río Irati. En este caso, Tesicnor generó el vuelo con el método *Generate* comentado anteriormente, y los pilotos responsables de la operación descargaron el fichero Flight Detail, para ver los POI y las restricciones de vuelo. Una vez comprobado la validez del vuelo, los pilotos descargan el fichero Flight Plan, que importan en *Mission Planner*, dónde pueden retocar el vuelo y añadir la configuración de la cámara que porta el dron.



(a) Google Earth Pro

(b) Mission Planner

Figura 23: Plan de vuelo río Irati

En la Figura 23(a), vemos el plan de vuelo en *Google Earth Pro*, en dónde una parte del recorrido se solapa con la restricción de vuelo debido a la cercanía del aeródromo de Lumbier. También aparecen con el icono de información los puntos de interés, correspondientes a puentes o presas, en dónde interesa que el dron descienda unos metros de altitud para visualizar mejor la zona.

En la Figura 23(b), vemos el plan de vuelo en *Mission Planner*, en dónde cada icono verde corresponde a un comando *WAYPOINT*, que indica las posiciones por las que el dron debe pasar. Al final del plan de vuelo se ha añadido el comando *RETURN_TO_LAUNCH*, para ordenar al dron a volver al punto *HOME* al finalizar el vuelo.

Generación de la ruta de vuelo

En este capítulo vamos a detallar el desarrollo llevado cabo para la generación automática de la ruta de vuelo a lo largo de cauces de ríos.

6.1. Contexto

Como vimos en la Sección 5.2, Tesicnor comunica a i3Code los puntos que conforman la ruta de vuelo, formada en su mayoría por puntos situados en el cauce del río. Por ejemplo, las líneas amarillas de la Figura 23, corresponden a rutas que sigue el dron en la misión de vuelo realizada en el río Irati. La obtención de dichos puntos se realiza manualmente, de manera que los encargados de definir el vuelo seleccionan cada uno de los puntos que lo forman. La automatización de este proceso ayudaría a reducir su el tiempo invertido en la definición de la ruta, lo cuál resulta de especial interés en operaciones realizadas en situaciones de emergencia.

6.2. Datos red hidrográfica

Una vez identificada la necesidad de generar automáticamente la ruta de vuelo del dron, hemos ideado un proceso para desarrollarla. El primer paso es la obtención y procesamiento de datos de la red hidrográfica, que sirvan de soporte para la autogeneración de waypoints. En lugar de definir cada uno de los waypoints de la ruta, podemos obtener la localización de cauces de ríos y seleccionar el tramo que nos interesa.

6.2.1. Obtención de datos

En la Subsección 4.2.1, ya obtuvimos una capa con los cauces de Navarra, la cuál filtrábamos para quedarnos con los ríos principales. Para este proceso, consideramos más adecuado generalizar el servicio a nivel nacional, por lo tanto, hemos obtenido la información sobre los cauces de toda España.

El Ministerio para la Transición Ecológica y el Reto Demográfico (MITECO) pone a disposición datos geográficos de distintos sectores. En el sector hidrográfico, expone tres capas vectoriales con los ríos completos clasificados según Pfafstetter modificado. El método de clasificación Pfafstetter es una propuesta de la Comisión Europea para la descripción de la topología de cuencas. En el documento [15], se explica el proceso llevado a cabo para la representación digital de toda la red fluvial de España, clasificando un total de 173.174 ríos completos y 333.583 cuencas.

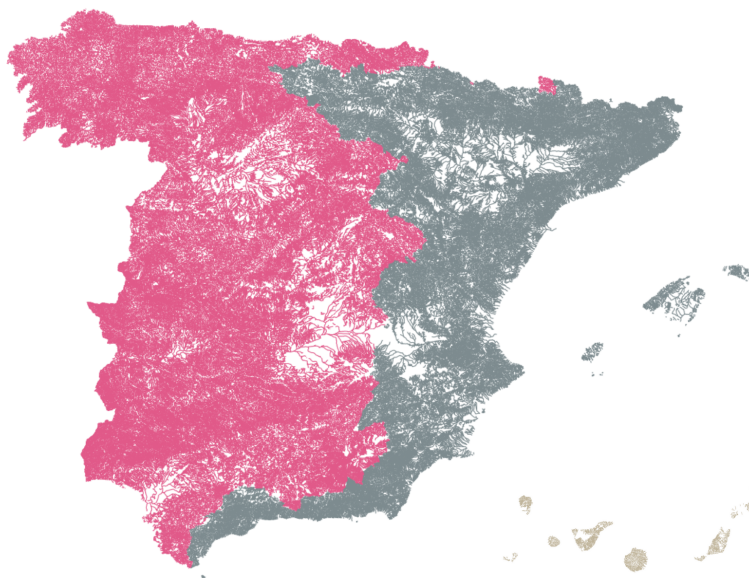


Figura 24: Capas ríos España

En la Figura 24, podemos observar las capas de ríos nacionales cargadas en la herramienta *QGIS*. Una capa para la mitad Oeste de la Península, otra para la mitad Este e Islas Baleares, y la última para las Islas Canarias. *QGIS* nos permite hacer la unión

de las tres capas, generando una única capa con todos los ríos de España, la cual podemos exportar a un archivo Shapefile. Finalmente, hemos comprimido a ZIP los archivos generados, reduciendo el tamaño un 49.78 %.

6.2.2. Preprocesamiento de datos

Al igual que en los procesamientos de datos que hemos ido viendo a lo largo del trabajo, vamos a emplear *GeoPandas* para manipular los datos obtenidos en la Sección 6.2. Podemos leer directamente el archivo ZIP con la capa de los ríos nacionales.

```
[1]: import geopandas as gpd
      gpd.read_file('C:\\EMERAL\\rios-nacionales.zip').set_crs(25830).to_crs(4326)
```

Como vemos, tenemos que especificar el Sistema de Referencia de Coordenadas o *Coordinate Reference System* (CRS) original de la capa. Como vimos en [11], el Sistema Europeo de Referencia Terrestre 1989 (EPSG:25830 / ETRS89) es el sistema de referencia geodésico oficial en España. Posteriormente aplicamos una transformación del CRS al Sistema Geodésico Mundial 1984 o World Geodetic System 1984 (EPSG:4326 / WGS84), que es el sistema de coordenadas universal, con el que finalmente nos quedamos.

```
[6]: from pandas as pd
      df = pd.DataFrame(rios.geom_type, columns=["geomType"])
      df.groupby(by="geomType")["geomType"].count()
```

```
[6]: geomType
      LineString          172696
      MultiLineString      17
      Name: geomType, dtype: int64
```

Hemos analizados las geometrías de la capa, vemos que la mayoría de ríos están formados por una sola línea, pero hay 17 que están formados por un conjunto de líneas.

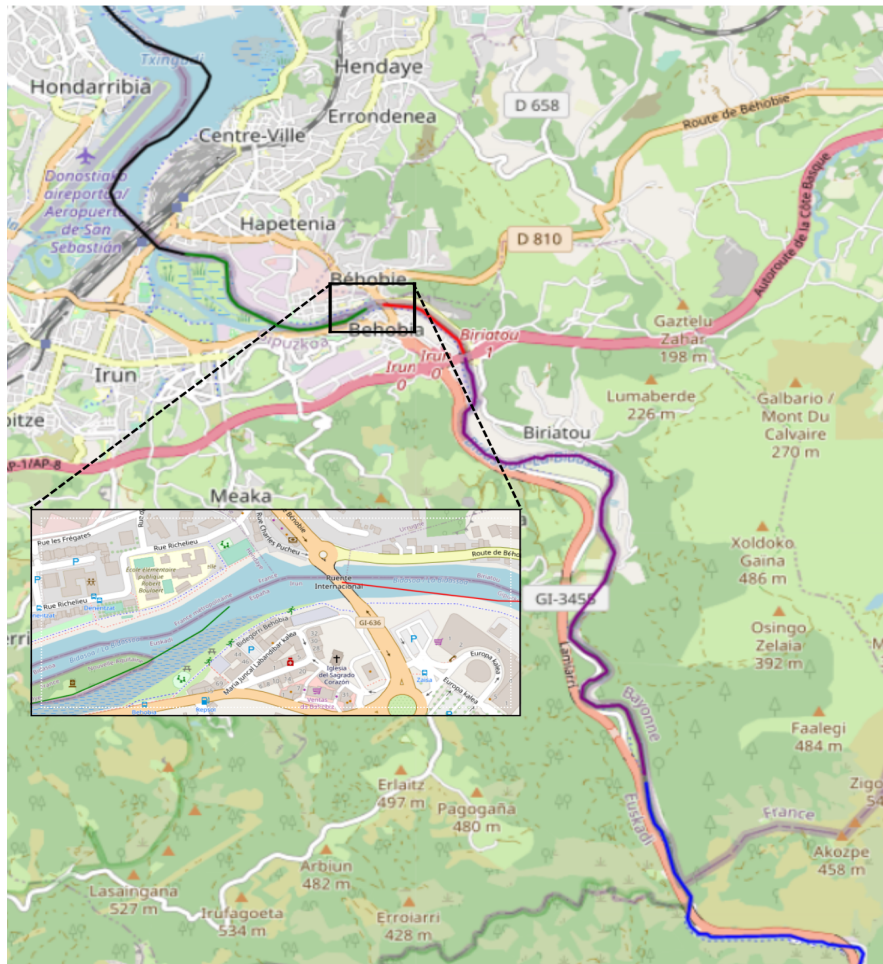


Figura 25: Tramos del río Bidasoa

Como vemos en la Figura 25, la geometría correspondiente al río Bidasoa viene dividida en líneas, representada cada una de un color distinto. Si nos acercamos al río a su paso por Behobia, vemos que falta un tramo de río.

Vamos a buscar la forma de agrupar los distintos tramos de los 17 ríos que están formados por multi-líneas.

```
[ ]: import geopandas as gdf
      from shapely.geometry import LineString
      # Obtener ríos multi-línea
      rios_multi = rios.loc[rios.geometry.type == "MultiLineString"]
```

```

for id, rio in rios_multi.iterrows():
    # Obtener coordenadas de las líneas
    rio_coord = [list(p.coords) for p in rio.geometry]
    # Crear una única línea con todas las coordenadas
    rio_geom = LineString([item for sublist in rio_coord for item in sublist])
    # Reemplazar la geometría
    rios.loc[[id], 'geometry'] = gpd.GeoSeries([rio_geom]).values

```

El tratamiento que hemos realizado ha sido completar los tramos que faltan uniéndolos con una línea recta. Para ello, hemos seleccionado los ríos formados por multi-líneas y hemos generado una única línea con las coordenadas de todos sus tramos.



(a) Bidasoa original



(b) Bidasoa procesado

Figura 26: Procesamiento geometría río Bidasoa

Finalmente, hemos sustituido la geometría original (multi-línea) por la geometría procesada (línea), tal y como vemos en el ejemplo del río Bidasoa de la Figura 26.

Con estos tratamientos damos por finalizado el preprocesamiento de los datos y entramos a ver el uso que les damos en el proceso de generación de la ruta de vuelo.

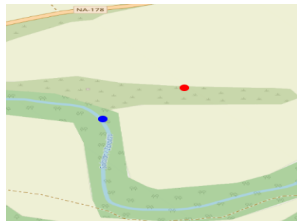
6.3. Proceso de generación de la ruta de vuelo

Una vez que tenemos la capa con los ríos nacionales procesada, podemos definir con más detalle el procedimiento para la generación automática de la ruta de vuelo, a través del cauce del río.

6.3.1. Descripción general del proceso

El proceso está formado por dos subprocesos:

- Búsqueda del punto más cercano



Como vemos en la imagen, a partir de un punto geográfico (punto rojo), situado en cualquier posición, buscamos el punto más cercano (punto azul) de la capa generada en Sección 6.2.

- Búsqueda del camino más corto



A partir de dos puntos geográficos situados en el cauce de ríos, obtenemos el camino más corto (línea roja) que une ambos puntos (puntos negros) a través

uno o más cauces, según sea necesario. Los datos que determinan el camino se obtienen de la capa generada en Sección 6.2.

6.3.2. Búsqueda del punto más cercano

```
[1]: from shapely.geometry import Point, LineString
      from shapely.ops import nearest_points
      rios_coords = rios.unary_union
      punto = Point(-1.275571808685088,42.65550922004611)
      punto_rio_mas_cercano = nearest_points(punto, rios_coords)[-1]
      coord = list(punto.coords) + list(punto_rio_mas_cercano.coords)
      linea = LineString(coord)
```

La propiedad *unary_union* de un *GeoDataFrame* devuelve una geometría que incluye a todas las geometrías, en este caso, obtenemos la unión de todos los ríos. A partir de esta unión y de un punto, obtenemos el punto del río más cercano mediante la función *nearest_points* de la librería *shapely*. Los objetos *GeoDataFrames* emplean objetos *shapely* para definir las geometrías y realizar operaciones geométricas.

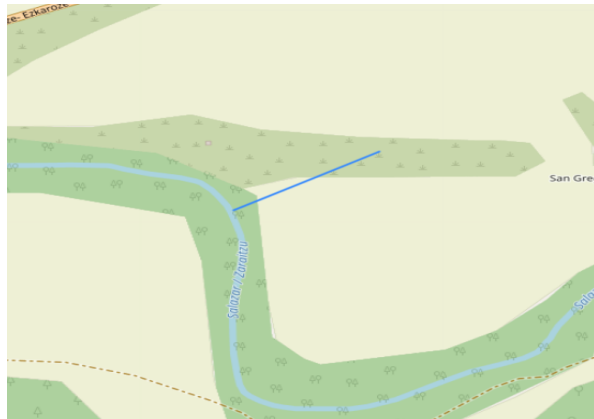


Figura 27: Punto más cercano del río Salazar

Finalmente creamos una geometría *LineString* con las coordenadas del punto inicial y el punto del río más cercano, la cuál podemos ver en la Figura 27 para el punto (-1.275571808685088, 42.65550922004611) y el río Salazar.

6.3.3. Búsqueda del camino más corto

En esta sección vamos a detallar la implementación del algoritmo de búsqueda del camino más corto. En primer lugar, vamos a definir el conjunto de datos de entrada empleados en el algoritmo.

Conjunto de datos de entrada

```
[1]: from shapely.geometry import Point, nearest_points
      p_ini = Point(-1.285, 42.689)
      p_fin = Point(-1.268, 42.654)
      p_ini_rio = nearest_points(p_ini, rios_coords)[-1]
      p_fin_rio = nearest_points(p_fin, rios_coords)[-1]
```

Partimos de dos puntos geográficos (p_ini y p_fin) y obtenemos los puntos de los ríos más cercanos a ellos (p_ini_rio y p_fin_rio). Necesitamos dos puntos situados en los cauces de algún río para establecer el camino más corto entre ellos.

```
[2]: import numpy as np
      from shapely.geometry import box
      p1 = np.array(p_ini_rio.coords)
      p2 = np.array(p_fin_rio.coords)
      p_coords = np.concatenate((p1, p2))
      grow_box = 0.09
      minx = p_coords[:, 0].min() - grow_box
      miny = p_coords[:, 1].min() - grow_box
      maxx = p_coords[:, 0].max() + grow_box
      maxy = p_coords[:, 1].max() + grow_box
      bbox = box(minx, miny, maxx, maxy)
```

El conjunto de datos con el que partimos es demasiado grande como para emplearlo por completo. Por lo tanto, definimos un *bounding_box* y seleccionamos solamente los tramos de ríos que se intersecten con él.

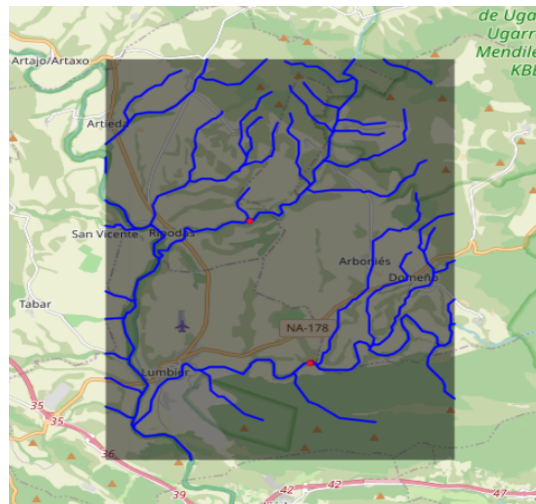


Figura 28: Bounding Box zona Lumbier

En la Figura 28 vemos el *bounding_box* generado con los dos puntos del ejemplo inicial (puntos rojos), situados en las inmediaciones de la localidad de Lumbier.

Podemos interpretar los puntos intersección junto a los puntos inicial y final como los nodos de un grafo, cuyas aristas son las líneas que los unen y los pesos su distancia.

```
[3]: # Recorrer los ríos por pares
for id1, rio1 in rios_bb.iterrows():
    for id2, rio2 in rios_bb.iterrows():
        if id2 > id1:
            # Obtener geometrías de los ríos
            rio1_geom = rio1.geometry
            rio2_geom = rio2.geometry
            # Comprobar intersección entre ríos
            if (rio1_geom.intersects(rio2_geom)):
                interseccion = rio1_geom.intersection(rio2_geom)
                # Añadir punto de intersección si no existe
                if interseccion not in puntos_interseccion:
                    puntos_interseccion.append(interseccion)
            # Guardar índices de los ríos
            rios_ids.append([id1, id2])
```


Recorremos los ríos por pares y comprobamos si existe intersección entre sus geometrías. En caso afirmativo, obtenemos y guardamos el punto de intersección, así como los índices de los ríos que intersectan.

```
[4]: nodos = puntos_interseccion.copy()
# Insertamos los puntos inicial y final al conjunto de nodos
nodos.insert(0, p_ini_rio)
nodos.append(p_fin_rio)
# Obtenemos los rios en los que se localizan los puntos inicial y final
rios_ids.insert(0, [rios_bb.distance(p_ini_rio).idxmin()])
rios_ids.append([rios_bb.distance(p_fin_rio).idxmin()])
```

Creamos un conjunto de nodos, incluyendo los puntos inicial y final, así como los ríos en los que se encuentran.

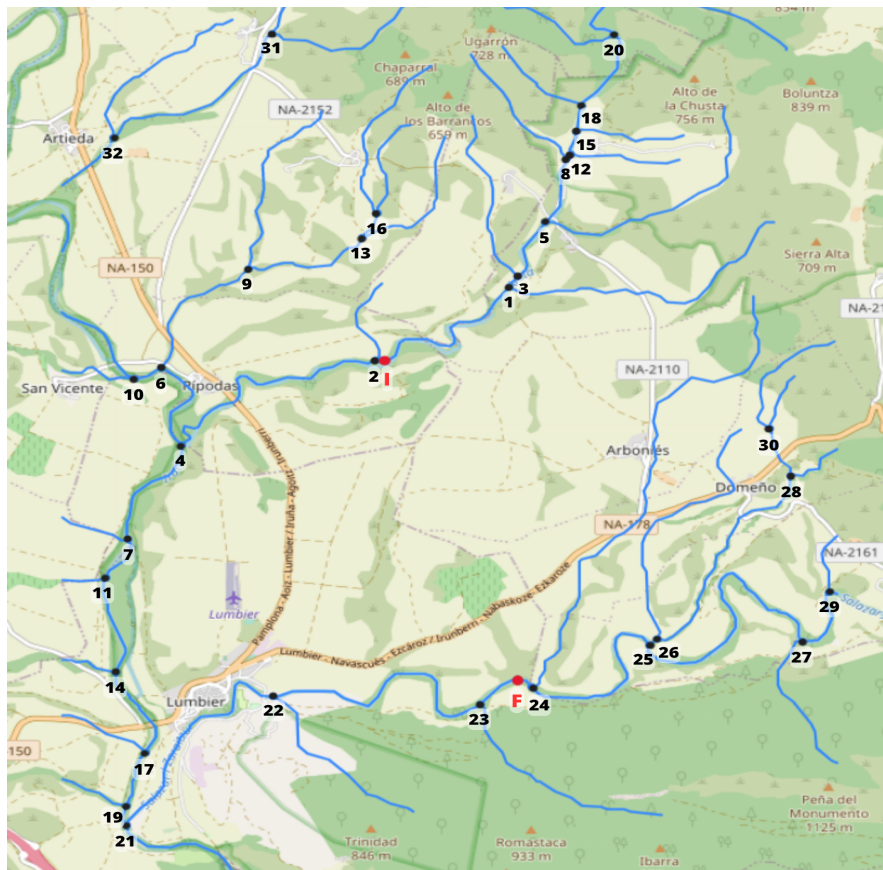


Figura 29: Ríos y puntos de intersección

En la Figura 29, vemos las intersecciones de los ríos incluidos en el *bounding_box*, marcadas con puntos negros numerados, y los puntos inicial (I) y final (F) marcados con puntos rojos.

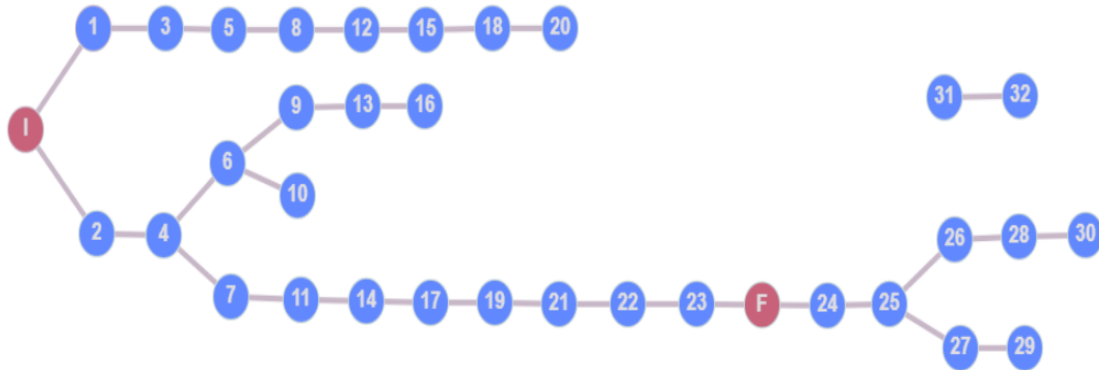


Figura 30: Grafo ríos zona Lumbier

El conjunto de datos lo podemos interpretar como el grafo de la Figura 30, en donde los nodos representan los puntos de intersección, junto con los puntos inicial y final, y las aristas representan los tramos de ríos que unen los puntos. Debemos tener en cuenta que el grafo generado no es un grafo conexo, es decir, cada par de vértices no tienen porqué estar conectados por un camino. Por lo tanto, no siempre encontraremos el camino entre cada par de puntos establecidos.

El último paso que nos falta para definir el conjunto de datos de entrada, es calcular los pesos del grafo Figura 30. A modo de simplificación, vamos a descartar algunos nodos para calcular los pesos. En concreto, los del grafo aislado (31 y 32) y los que tienen un nivel de profundidad superior al punto final (del 24 al 30).

Una vez seleccionados los nodos que vamos a emplear, calculamos el peso entre cada par de vértices adyacentes (unidos por una arista), es decir, la distancia entre cada par de puntos unidos por un río.

```

[5]: # Recorremos los puntos/nodos por pares
for id1, nodo1 in enumerate(nodos):
    for id2, nodo2 in enumerate(nodos):
        if id1 != id2:
            # Obtenemos el id del río en el que se encuentran ambos puntos
            for rio_comun_id in list(set(rios_ids[id1]) & set(rios_ids[id2])):
                rio = rios_bb.iloc[rio_comun_id].geometry
                # Calculamos la distancia desde el inicio del río cada punto
                dist_ini = rio.project(nodo1.geometry)
                dist_fin = rio.project(nodo2.geometry)
                # Obtenemos el tramo del río y guardamos su longitud
                tramo_rio = substring(rio, dist_ini, dist_fin)
                dist_tramo = round(tramo_rio.length)
                distancias[id1, id2] = dist_tramo
                distancias[id2, id1] = dist_tramo
            else:
                # La distancia de un nodo con sí mismo es 0
                distancias[id1, id2] = 0

```

0	2652	93	2847	4085	3823	∞	∞	4819	∞	∞	∞	4915	∞	5267	∞	5631	∞	6913	∞	∞	∞
2652	0	2745	195	6737	1171	∞	∞	2167	∞	∞	2263	∞	∞	2615	∞	2978	∞	4261	∞	∞	∞
93	2745	0	2941	3992	3916	∞	∞	4912	∞	∞	5008	∞	∞	5360	∞	5724	∞	7006	∞	∞	∞
2847	195	2941	0	6932	975	∞	∞	1972	∞	∞	2067	∞	∞	2419	∞	2783	∞	4065	∞	∞	∞
4085	6737	3992	6932	0	7908	1585	1735	8904	∞	2037	2463	8999	∞	3903	9352	5321	9715	10997	6460	∞	∞
3823	1171	3916	975	7908	0	∞	∞	996	∞	∞	1092	∞	∞	1444	∞	1808	∞	3090	∞	∞	∞
∞	∞	∞	∞	1585	∞	0	3320	∞	2218	451	4048	∞	∞	5488	∞	6907	∞	7712	∞	8045	∞
∞	∞	∞	∞	1735	∞	3320	0	∞	∞	3772	728	∞	∞	2168	∞	3586	∞	4391	∞	4725	∞
4819	2167	4912	1972	8904	996	∞	∞	0	∞	∞	96	∞	∞	448	∞	811	∞	2094	∞	∞	∞
∞	∞	∞	∞	∞	∞	2218	∞	∞	0	∞	∞	1968	∞	∞	∞	∞	∞	∞	∞	∞	∞
∞	∞	∞	∞	2037	∞	451	3772	∞	∞	0	4500	∞	∞	5940	∞	7358	∞	8163	∞	8496	∞
∞	∞	∞	∞	2463	∞	4048	728	∞	∞	4500	0	∞	∞	1440	∞	2858	∞	3663	∞	3997	∞
4915	2263	5008	2067	8999	1092	∞	∞	96	∞	∞	0	∞	∞	352	∞	716	∞	1998	∞	∞	∞
∞	∞	∞	∞	∞	∞	∞	∞	∞	1968	∞	∞	0	∞	433	∞	∞	∞	∞	∞	∞	∞
∞	∞	∞	∞	3903	∞	5488	2168	∞	∞	5940	1440	∞	∞	0	∞	1418	∞	2223	∞	2557	∞
5267	2615	5360	2419	9352	1444	∞	∞	448	∞	∞	352	∞	∞	0	∞	364	∞	1646	∞	∞	∞
∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	433	∞	∞	∞	∞	∞	∞	∞	∞	∞
∞	∞	∞	∞	5321	∞	6907	3586	∞	∞	7358	2858	∞	∞	1418	∞	0	∞	805	∞	1139	∞
5631	2978	5724	2783	9715	1808	∞	∞	811	∞	∞	716	∞	∞	364	∞	∞	0	1282	∞	∞	∞
∞	∞	∞	∞	6126	∞	7712	4391	∞	∞	8163	3663	∞	∞	2223	∞	805	∞	334	∞	∞	∞
6913	4261	7006	4065	10997	3090	∞	∞	2094	∞	∞	1998	∞	∞	1646	∞	1282	∞	0	∞	∞	∞
∞	∞	∞	∞	6460	∞	8045	4725	∞	∞	8496	3997	∞	∞	2557	∞	1139	∞	334	∞	3495	7452
∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	0	∞	3956	4652
∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	7452	3956
∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	8147	4652
∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	695	0
∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	8147	4652
∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	695	0

distancias =

En la matriz de distancias, cada fila representa la distancia en metros de un punto respecto al resto de puntos, es decir, la posición (i, j) representa la distancia del tramo del río que va desde el punto i hasta el punto j . Por lo tanto, la distancia será 0 metros cuando $i = j$, referente a la distancia de un punto consigo mismo, y será infinito cuando no exista conexión entre punto i y el punto j , es decir, cuando los puntos no se encuentran en el mismo río.

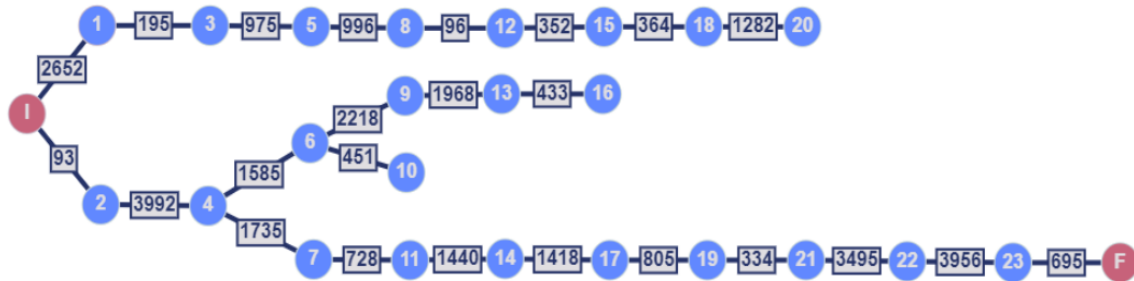


Figura 31: Grafo ponderado zona Lumbier

A partir de la matriz de distancias, podemos generar el grafo ponderado de la Figura 31. Este grafo va a servir como entrada al algoritmo del camino más corto.

Problema del camino más corto en teoría de grafos

En la teoría de grafos, el problema del camino más corto es el problema que consiste en encontrar un camino entre dos vértices o nodos, de tal manera que la suma de los pesos de las aristas que lo constituyen sea mínima.

Hemos escogido el algoritmo de Dijkstra, llamado así por su creador Edsger Wyde Dijkstra, nacido en Países Bajos en 1930, físico teórico y programador en el Centro Matemático de Ámsterdam. Dijkstra publicó el algoritmo en el año 1959. El algoritmo de Dijkstra determina el camino más corto desde un vértice hacia el resto de vértices del grafo. En [16] se emplea el algoritmo para calcular el camino más corto por cauces de ríos, pero en este caso se considera la dirección de la corriente, ya que su objetivo es determinar una ruta para el transporte de mercancía o personas.

Entrada: G : Grafo

Salida: P : Vector[1,N], predecesores, D : Vector[1,N], Distancias mínimas, con

N el n^o de nodos de G

$D = \{\}$; // Distancias mínimas

$P = \{\}$; // Conjunto de predecesores

$Q = \{1,2,\dots,n\}$; // Conjunto del nodos de G

para cada *vértice* $v \in G$ **hacer**

$D[v] = \infty$;
 $P[v] = \emptyset$;

fin

mientras $Q \neq \emptyset$ **hacer**

$u =$ nodo de Q con la mínima distancia; // Con más de uno, aleatorio
 eliminar u de Q ;

para cada *vertice adyacente* v de u **hacer**

si $D[v] > D[u] + \text{peso_arista}(u, v)$ **entonces**
 $D[v] = D[u] + \text{peso_arista}(u, v)$;
 $P[v] = u$;
 fin

fin

fin

devolver D, P ;

Algoritmo 1: Algoritmo Dijkstra.

El algoritmo 1 tiene como entrada el grafo ponderado G y devuelve como salida el vector P con los predecesores de cada nodo y el vector D con las distancias mínimas. El procedimiento que sigue es el siguiente:

1. Inicializar las distancias mínimas a infinito y los predecesores a valores nulos.
2. Obtenemos el vértice u del grafo G cuya distancia al nodo antecesor sea mínima. Si hay más de una distancia mínima con el mismo valor, se escoge aleatoriamente un nodo con distancia mínima. Esto sucede en la primera iteración, en la no hay antecesor con el que comparar. También se puede determinar como entrada el nodo origen, que sería el primer antecesor.
3. Recorremos los vértices v adyacentes al vértice u . Si la distancia del camino hasta llegar a v pasando por u es menor que la encontrada hasta ese momento, actualizamos la distancia mínima y el predecesor, que en este caso sería u .
4. Descartamos u para las siguientes iteraciones y volvemos a repetir el procedimiento desde el punto 2, hasta tratar todos los nodos del grafo G .

Vamos a aplicar el algoritmo 1 para el ejemplo de la Figura 29. Para ello, hemos recorrido cada nodo comenzando por el nodo inicial (I), hasta llegar al nodo final (F). Por cada nodo, hemos buscado el camino más corto desde el nodo inicial. Las iteraciones llevadas a cabo se muestran en la Tabla 6.2.

Disponemos los nodos como filas y las iteraciones como columnas. Cada celda puede contener tres tipos de valores:

- * para determinar que el nodo ya no se va a continuar tratando.
- ∞ para determinar que no existe camino.
- Par $(dist_min, predecesor)$, para determinar cuál es la distancia mínima para llegar al nodo y cuál es el predecesor en el camino más corto. En negrita aparecerá el valor definitivo.

Según la Tabla 6.2 vemos que el camino más corto del nodo I al nodo F tiene una distancia de 18.691 metros. Recorriendo a la inversa los antecesores, desde el nodo F al nodo I , podemos obtener el camino más corto entre ambos nodos:

$$F \leq 23 \leq 22 \leq 21 \leq 19 \leq 17 \leq 14 \leq 11 \leq 7 \leq 4 \leq 2 \leq I$$

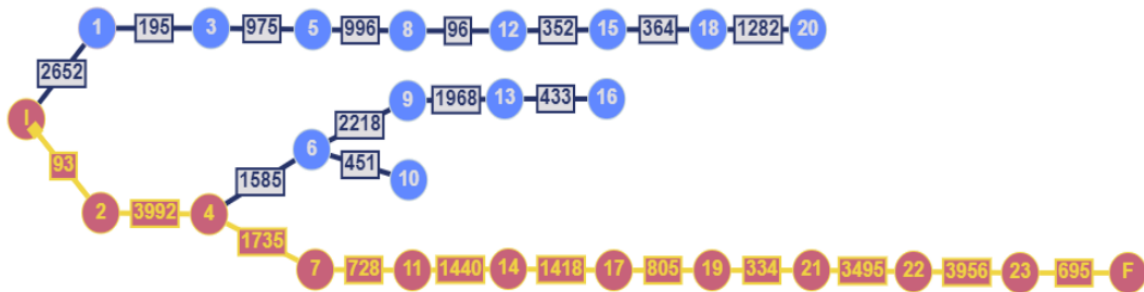


Figura 32: Camino más corto en el grafo



Figura 33: Camino más corto ejemplo Lumbier

En la Figura 32 y en la Figura 33 podemos observar el resultado obtenido del algoritmo del camino más corto. La primera resalta el camino más corto en el grafo y la segunda en el mapa.

Las redes fluviales tienen una disposición ramificada, por lo que raramente generan grafos con ciclos, es decir, caminos cerrados en los que no se repiten ningún vértice, a excepción del primero, que marca el inicio y fin del camino. Por lo tanto, vemos que el camino mínimo normalmente es el único camino que existe entre el punto inicial y final.

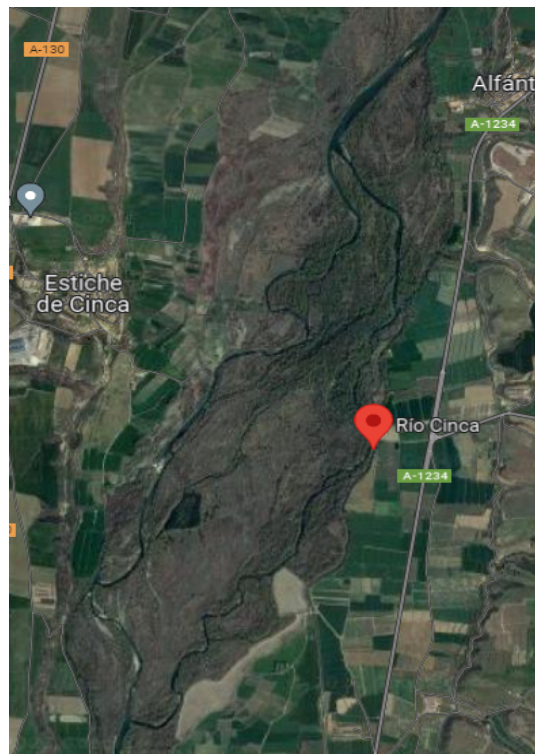


Figura 34: Ejemplo río trezado (Río Cinca)

La solución aportada contempla la existencia de ciclos, por ejemplo para calcular el camino más corto sobre ríos trezados, como por ejemplo el de la Figura 34, correspondiente al río Cinca, localizado en las provincias de Huesca y Lérida. La capa generada en la Sección 6.2, unifica los distintos canales del río Cinca en un sólo cauce, por los que no es posible generar un grafo con ciclos con los datos que tenemos.

6.4. Despliegue

En esta sección vamos a ver cómo hemos incluido la generación automática de la ruta en la plataforma web comentada en la Subsección 5.2.2.

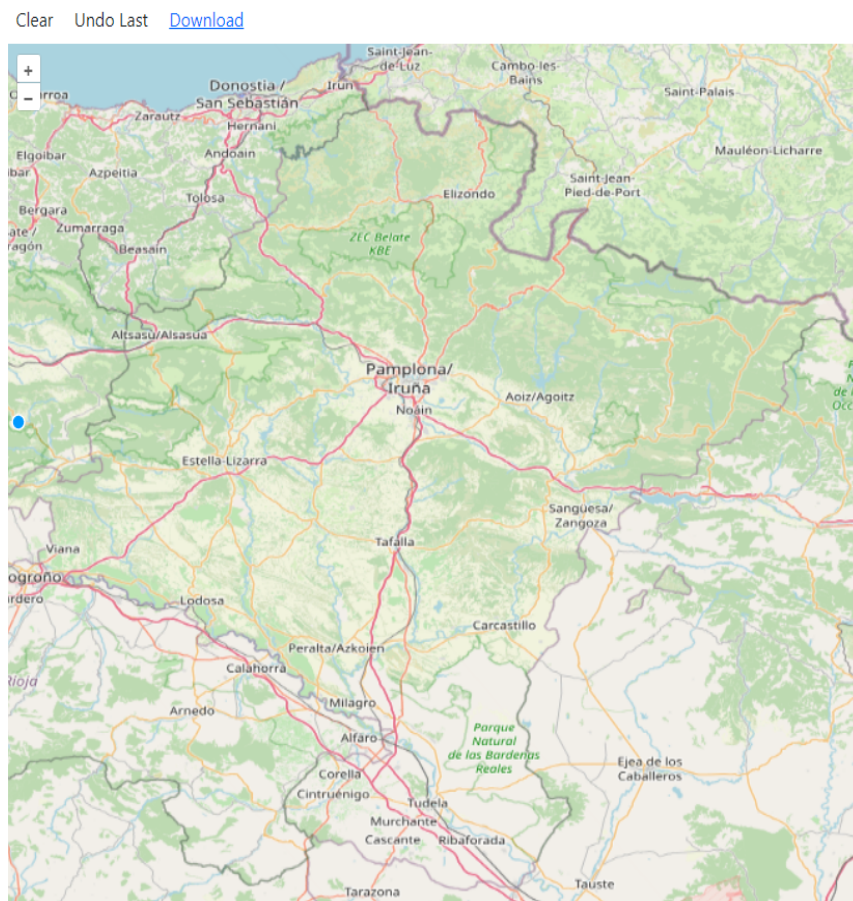


Figura 35: Mapa interactivo web EMERAL

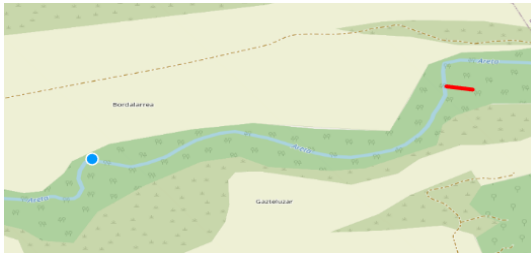
Hemos creado el mapa interactivo de la Figura 35 con el que establecer los puntos de entrada vistos en la Subsección 6.3.1. Hemos empleado la librería *OpenLayers* de *JavaScript* para su implementación.



(a) Selección 1^{er} punto



(b) Recta al punto más cercano



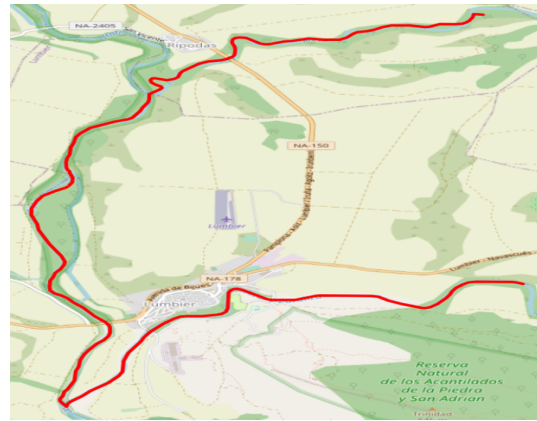
(c) Selección 2^o punto



(d) Camino más corto entre el 1^{er} y 2^o punto



(e) Selección 3^{er} punto



(f) Camino más corto entre el 2^o y 3^{er} punto

Figura 36: Pasos generación automática de la ruta

En la Figura 36 vemos las interacciones con el mapa (36(a), 36(c) y 36(e)), y los resultados que vamos obteniendo con cada interacción (36(b), 36(d) y 36(f)). El resultado final es igual a la ruta a la que hemos llegado en la Figura 33. En primer lugar obtenemos la recta desde el punto inicial de la ruta al punto del río más cercano, y en segundo lugar, vamos seleccionando los puntos del río por dónde nos interesa generar la ruta, hasta llegar al punto final de la misma.

Podemos observar que la ruta está en las inmediaciones de aeródromo, que tal y como vimos en el Capítulo “Identificación de restricciones de vuelo”, corresponde a una zona restringida al vuelo de drones. Por lo tanto, hemos incluido esta información en el proceso de generación de la ruta, para indicar al piloto que la ruta atraviesa una zona restringida para el vuelo de drones.

Vamos a realizar otro ejemplo para comprobar esta funcionalidad.

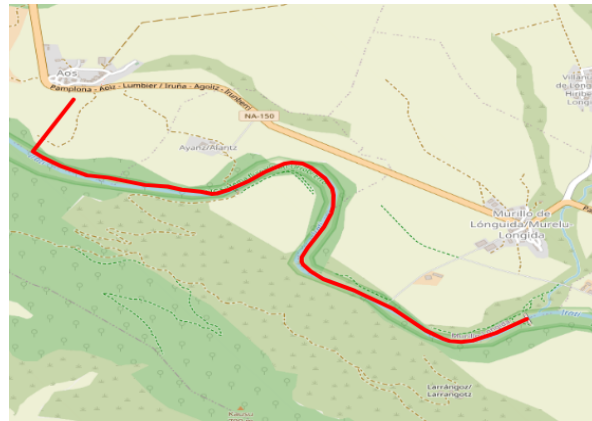


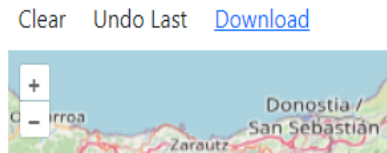
Figura 37: Ruta por río Irati desde Aos hasta Murillo de Lónguida

En la Figura 37 vemos la ruta generada por el cauce del río Irati, desde el municipio de Aos hasta Murillo de Lónguida. En este tramo, la ruta no ha entrado en ninguna zona restringida.



Figura 38: Ruta por río Irati desde Aos hasta Artajo

En la Figura 38 vemos la continuación de la ruta generada por el cauce del río Irati, hasta el municipio de Artajo. Podemos observar que al llegar a este municipio entramos en una zona resaltada en azul, que alejándonos en el mapa, podemos comprobar que corresponde a una zona restringida por la alerta del Aeródromo de Lumbier.



Si nos fijamos en el menú superior del mapa, vemos que tenemos la posibilidad de eliminar toda la ruta generada, o eliminar solamente el último tramo generado. Por lo que podemos volver a la situación de la Figura 37 una vez estamos en la situación de la Figura 38.

También tenemos la opción de descargar la ruta en formato WAYPOINT con los comandos de vuelo, del mismo modo que descargábamos el plan de vuelo mediante la opción Flight Plan que vimos en Subsección 5.2.2.

Con este servicio, damos por finalizada la aportación propuesta a pilotos de drones, en cuanto a planificación de vuelo se refiere.

Conclusiones y líneas futuras

7.1. Conclusiones

Los drones tienen la capacidad de dotar a los servicios de emergencias de un despliegue rápido, sin riesgos humanos, sin emisiones contaminantes y con una mayor eficiencia económica de los recursos públicos. Sin embargo, los drones empleados actualmente por los cuerpos de emergencia no pueden apoyar de forma efectiva la resolución de inundaciones.

La planificación de la misión de vuelo del dron se realiza de forma manual. Es decir, la mayoría de los drones vuelan de forma automática, siguiendo la ruta pre-programada por el piloto. Planificar vuelos de decenas de kilómetros de forma manual en escenarios tan complejos y dinámicos, como los que encontramos en situaciones de emergencia, elimina la posibilidad de usar drones. Dotar a los pilotos de herramientas que automaticen parte de la planificación de vuelo, permitirá aumentar la capacidad de respuesta de los drones ante este tipo de situaciones, de tal manera que suponga una alternativa eficaz a los sistemas de despliegue aéreo actuales.

Además, los drones no están integrados en espacio aéreo, no pueden volar junto con otras aeronaves que estén volando en la misma zona (p. ej. Helicópteros sanitarios). Existe una legislación a nivel europeo que regula la integración de los drones con las aeronaves tripuladas, pero que actualmente se encuentra en fase de implementación de los servicios iniciales, dentro del ecosistema U-Space.

Por lo tanto, vemos como los drones van cobrando cada vez más protagonismo en muchos sectores, hasta el punto que formarán parte de nuestra vida cotidiana. En esta transición al uso de drones, también se encuentran los servicios de emergencias, que mediante proyectos como EMERAL, se avanza en conseguir dicho objetivo.

7.2. Líneas futuras

El proyecto EMERAL finaliza a finales de 2023, por lo que seguiremos desarrollando las partes que hemos visto en este trabajo. A continuación, enumeramos las distintas vías de desarrollo planificadas hasta finalizar el proyecto.

- Servicio de información meteorológica
 1. Generar gráficas de los datos meteorológicos e hidrométricos, en las que mostrar líneas de tendencias y umbrales de alerta.
 2. Incluir datos de predicciones y buscar nuevas fuentes de información.
 3. Mostrar datos meteorológicos e hidrométricos mientras se genera la ruta desde el mapa interactivo de la web EMERAL.
- Generación automática de la ruta de vuelo
 1. Comparar la eficiencia de distintos algoritmos de búsqueda del camino más corto.
 2. Emplear capas vectoriales con distinta información, por ejemplo, capas con información de pistas de montaña y senderos. Con el objetivo de generalizar el uso del servicio de generación de ruta mas corta.
- Despliegue de los servicios

Realizar pruebas de estrés para determinar los límites del sistema en el que está desplegado los servicios y contemplar la necesidad de mejorarlo.

Bibliografía

- [1] B. M. Balaguer, “El futuro de los drones - centum digital.” <https://centum.com/el-futuro-de-los-drones/>.
- [2] “Primer vuelo de dron más allá de la línea visual sobre una red eléctrica en España.” <https://www.youtube.com/watch?v=iKDVpcKElyQ&t=5s>.
- [3] La Comisión Europea, “REGLAMENTO DELEGADO (UE) 2019/945 DE LA COMISIÓN sobre los sistemas de aeronaves no tripuladas y los operadores de terceros países de sistemas de aeronaves no tripuladas,” Marzo 2019.
- [4] La Comisión Europea, “REGLAMENTO DELEGADO (UE) 2020/1058 DE LA COMISIÓN por el que se modifica el Reglamento Delegado (UE) 2019/945 en lo que respecta a la introducción de dos nuevas clases de sistemas de aeronaves no tripuladas,” Abril 2020.
- [5] La Comisión Europea, “REGLAMENTO DE EJECUCIÓN (UE) 2019/947 DE LA COMISIÓN relativo a las normas y los procedimientos aplicables a la utilización de aeronaves no tripuladas,” Mayo 2019.
- [6] Ministerio De Transportes, “Resolución de la dirección de la Agencia Estatal de Seguridad Aérea por la que se aprueban escenarios estándar nacionales (STS-ES) para operaciones de UAS en la categoría «específica»,”
- [7] “Listado de Empresas Certificadas por AESA en relación a los Servicios de Navegación Aérea.” https://www.seguridadaerea.gob.es/sites/default/files/listado_proveedores_ats_ais_cns_certificadas.pdf.
- [8] La Comisión Europea, “REGLAMENTO DE EJECUCIÓN (UE) 2021/664 DE LA COMISIÓN sobre un marco regulador para el U-Space,” Abril 2021.
- [9] “GUÍA SOBRE REQUISITOS Y LIMITACIONES AL VUELO DE UAS EN FUNCIÓN DEL LUGAR DE OPERACIÓN (ZONAS GEOGRÁFICAS DE UAS).” https://www.seguridadaerea.gob.es/sites/default/files/UAS-OPS-DT01_v6_Zonas_geograficas.UAS_.pdf.
- [10] “ENAIRES Drones.” <https://drones.enaire.es/>.

- [11] Jon Mikel Merino Ruiz, “Aplicaciones IA en los Sistemas de Información Geográfica,” Septiembre 2022.
- [12] “Documentación dinámica de los recursos de la API de AEMET.” <https://opendata.aemet.es/dist/index.html?>
- [13] “Sitio web del Servicio de Información Meteorológica y de Aforo.” <https://idena-api.i3code.net/>.
- [14] “ArcGIS REST Services of Drones.” https://servais.enaire.es/insigniad/rest/services/NSF_APP/Drones_APP_V1/MapServer.
- [15] “TRAMOS DE RÍOS DE ESPAÑA CLASIFICADOS SEGÚN PFAFS-TETTER MODIFICADO.” <https://sig.mapama.gob.es/Docs/PDFServicios/RiosPfafs.pdf>.
- [16] N. Cadieux, M. Kalacska, O. T. Coomes, M. Tanaka, and Y. Takasaki, “A python algorithm for shortest-path river network distance calculations considering river flow direction,” *Data 2020, Vol. 5, Page 8*, vol. 5, p. 8, 1 2020.