



UNIVERSIDAD DE VALLADOLID



**ESCUELA TÉCNICA SUPERIOR
DE INGENIEROS DE TELECOMUNICACIÓN**

TRABAJO FIN DE GRADO
GRADO EN INGENIERIA DE TECNOLOGÍAS ESPECÍFICAS
DE TELECOMUNICACIÓN.
TELEMÁTICA

DESARROLLO DE APLICACIÓN WEB BASADA EN
INFRAESTRUCTURAS DE DATOS ESPACIALES PARA LA
EVALUACIÓN DE ZONAS SEGURAS DE VUELO CON
DRONES

AUTOR: DANIEL VERDNIK CORREDERA
TUTOR: JUAN PABLO DE CASTRO

22 de septiembre de 2020

TITULO: Desarrollo de aplicación web basada en infraestructuras de datos espaciales seguras de vuelo con dron.

AUTOR: Daniel Verdnik Corredera

TUTOR: Juan Pablo de Castro Fernández

DEPARTAMENTO: TSCIT TSC

Miembros del Tribunal

PRESIDENTE: Juan Pablo de Castro Fernández

SECRETARIO: Luisa Regueras Santos

VOCAL: María Jesús Verdú Pérez

FECHA DE LECTURA:

CALIFICACIÓN:

RESUMEN DEL PROYECTO

El aumento del uso de los drones y su capacidad de crecimiento da lugar a nuevos retos que afrontar y nuevos problemas, esto ha dado lugar a la creación de normativas para controlar su uso. Debido al aumento de incidentes debido al uso recreativo de estos dispositivos se ha buscado una posible solución a este problema mediante el desarrollo de una aplicación web. Esta aplicación web está basada en el uso de tecnologías IDE (Infraestructuras de Datos Espaciales) para emplear tratar y mostrar información geográfica de forma comprensible a un usuario para conocer las diferentes zonas en las que existen restricciones al vuelo para este tipo de dispositivos.

Contenido

INTRODUCCIÓN.....	5
HISTORIA DE LOS DRONES.....	6
AERONAVES NO TRIPULADAS EN LA ACTUALIDAD.....	7
USO EN EVENTOS	7
SITUACIONES DE EMERGENCIA.....	7
TOPOGRAFIA	8
AGRICULTURA.....	8
INCENDIOS FORESTALES	8
ACTIVIDADES INDUSTRIALES	9
RECREATIVO	9
EVOLUCION DE LOS DRONES.....	9
DRONES EN EL SECTOR INDUSTRIAL.....	13
INCENDIOS FORESTALES	13
AGRICULTURA.....	13
CONSTRUCCIÓN Y CARTOGRAFÍA.....	14
VIGILANCIA EN CARRETERAS.....	14
PREVENCIÓN DE CONTAGIO.....	14
BÚSQUEDAS Y RESCATES.....	15
FUTURO DE LOS DRONES.....	15
PROBLEMAS RELACIONADOS CON EL USO DE DRONES	18

<i>EASYDRON</i>	20
OBJETIVOS	20
ESTUDIO DE MERCADO	22
RUNAWAY HD.....	22
UAV FORECAST	22
KITTY HAWK.....	22
DRONE BUDDY	23
ENAIRE.....	23
GEO ZONE MAP	24
AIRMAP	24
<i>ANÁLISIS Y DISEÑO</i>	26
ANÁLISIS	26
DISEÑO	29
INTERFAZ	29
PROCESAMIENTO DE LOS DATOS.....	30
ACCIONES DEL USUARIO	32
<i>DESARROLLO DE LA APLICACIÓN</i>	35
SERVIDOR	35
DATOS	35
BASE DE DATOS.....	41
SERVIDOR DE MAPAS	47
CLIENTE	52
SOPORTE WEB.....	52
<i>CONCLUSIONES</i>	98
<i>Líneas Futuras</i>	98
<i>Referencias</i>	100

INTRODUCCIÓN

En la actualidad los drones han pasado a ser una nueva herramienta con un gran potencial en diferentes campos, lo que supone un gran incremento de dispositivos y personas que los utilizan. En el caso de España en septiembre de 2019 el número de operadores de dron registrados supera los 3000, por otra parte, existe también un gran número de drones usados de forma recreativa que no requiere una licencia de vuelo y por tanto el número exacto es desconocido (Hernández 2019). Este tipo de tecnología ha permitido el desarrollo de nuevas actividades laborales y nuevas formas de entretenimiento lo que resulta atractivo para desarrollar el sector en diferentes apartados.

Un dron según la RAE se define como una aeronave no tripulada, podemos ampliar esta definición atendiendo al Real Decreto 1036/2017 («BOE-A-2017-15721.pdf» s. f.), donde nos encontramos con la definición de dron como una aeronave pilotada por control remoto distinguiéndose en diferentes tipos en función del tipo de propulsión (Figura 1):

- Rotor único: aquella aeronave no tripulada con una única hélice como forma de propulsión
- Multirrotor: aeronave no tripulada con dos o más rotores.
- Ala fija: propulsión a través de alas fijas
- Híbridos: aeronave que consta tanto de ala fija como de rotores

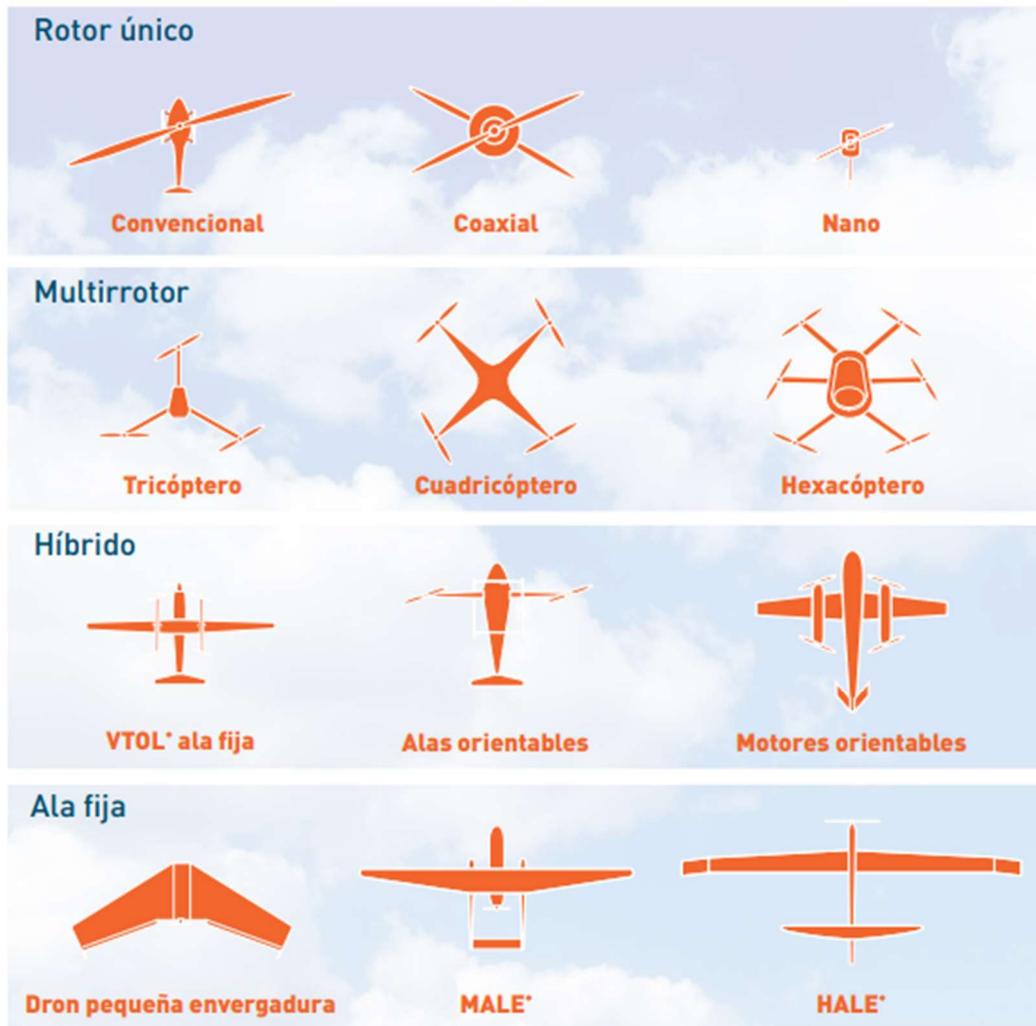


Figura 1: Tipos de dron en función de la propulsión

HISTORIA DE LOS DRONES

El dron que conocemos hoy en día no es un invento actual, sino que desde el siglo XIX ya se comenzó a desarrollar diferentes modelos. Estos modelos no estaban capacitados con la tecnología actual, pero se trataba de aeronaves no tripuladas utilizadas la mayoría en el ámbito bélico. Estas aeronaves se utilizaban para obtener información estratégica del enemigo o para realizar ataques cargándolas con explosivos. Un ejemplo fue el lanzamiento por parte de los austriacos de globos aerostáticos no tripulados cargados con bombas en julio de 1849 sobre la ciudad de Venecia. El desarrollo de estas aeronaves tuvo un mayor crecimiento con la revolución del sector de las telecomunicaciones, en concreto con el descubrimiento de las ondas de radio. Nikola Tesla fue el primero que mediante un radio control

puedo manejar a distancia un pequeño barco (Nikola 1898). Este avance supuso la creación de nuevas aeronaves no tripuladas también usadas en conflictos bélicos y estrategias contra los enemigos.

Durante la guerra de Vietnam se produjo un gran avance en el sector de los drones ya que comenzó una gran investigación para el desarrollo de aeronaves no tripuladas de vigilancia. Pero no fue hasta los años 80-90 cuando el dron como hoy lo conocemos comenzó a tomar forma gracias a los avances en sistemas electrónicos de control y la computación.

Durante estos años la mayoría de las aeronaves no tripuladas se han desarrollado con carácter militar, bien sea para reconocimiento geográfico o para la realización de ataques a distancia. En la actualidad estos dispositivos han empezado a utilizarse y desarrollarse pensando en actividades civiles, no solo militares, lo cual genera un gran campo de estudio.

AERONAVES NO TRIPULADAS EN LA ACTUALIDAD

Actualmente debido a la mejora tecnológica y el aumento de producción de este tipo de aeronaves junto a la reducción de precio de estas ha permitido que estas lleguen a diferentes sectores en el ámbito civil y el uso en nuevas actividades. Algunos de los usos actuales son:

USO EN EVENTOS

En la actualidad es muy común ver el uso de Drones en eventos deportivos («Uso de drones en eventos deportivos: el futuro de las transmisiones» 2020), lo que permite una mayor capacidad de retransmisión. Gracias al pequeño tamaño de estos dispositivos y la gran calidad de las cámaras que llevan incorporadas pueden grabar tomas y ángulos que con las cámaras tradicionales serían imposibles de conseguir lo que hace que el espectador desde su casa pueda disfrutar más del evento.(«Uso de drones en eventos deportivos: el futuro de las transmisiones» 2020)

SITUACIONES DE EMERGENCIA

Quizá uno de los usos más importante que puede darse a este tipo de aeronaves sea en situaciones de emergencia. Estos dispositivos permiten llegar a zonas de difícil

acceso para llevar elementos de rescate, contactar con personas en peligro o realizar búsquedas. Otro uso importante puede ser el transporte de órganos para realizar un trasplante como se realizó por primera vez desde el Hospital de St. Agnes en Baltimore al Centro Médico de la Universidad de Maryland, en abril de 2019 (López 2019). Realizando el transporte de esta manera se reduce considerablemente el tiempo debido a que no hay restricciones de tráfico, atascos... y permite llevar rápidamente y en buenas condiciones el órgano hasta el paciente lo que puede salvar vidas.

TOPOGRAFIA

En el ámbito topográfico y geográfico permite la realización de vuelos en los que se obtienen fotografías del territorio, acceder a zonas peligrosas como pueden ser los volcanes, permitiendo a los expertos obtener información, analizarla y realizar las investigaciones más rápidamente y reduciendo el coste. Con esta información puede predecir fenómenos naturales como puede ser la erupción de un volcán, o incluso predecir la ruta que seguirá un huracán analizando la temperatura y humedad dentro del ojo («Drones para estudiar huracanes -» 2017).

AGRICULTURA

Con la utilización de aeronaves no tripuladas con múltiple tecnología incorporada aparece el termino agricultura de precisión. Esto permite mejorar la productividad y reducir en impacto medioambiental. Mediante el uso de infrarrojos colocados en las aeronaves se puede analizar el estado de los cultivos, la fertilidad del suelo o el rendimiento del cultivo en grandes zonas de forma fácil. También se puede realizar control y extinción de plagas, facilita la aplicación de pesticidas sin necesitar de contar con un gran equipo de personas que lo realicen manualmente, además se puede realizar el tratamiento únicamente en las zonas afectadas, gracias a la detección GPS (Córdoba 2020).

INCENDIOS FORESTALES

En este sector los drones son empleados para la detección temprana de incendios, incluso prevenir y detectar posibles incendios antes de que se produzcan activando las medidas oportunas en cada caso, reduciendo los daños medioambientales y los

costes de extinción. Como en los demás usos también hay que tener en cuenta que evita que las personas no corran tanto riesgo (Geoinnova 2017).

ACTIVIDADES INDUSTRIALES

En la actualidad en la industria se utilizan estas aeronaves para el control de fugas, detección de emisiones tóxicas. Para ello se utilizan grabaciones de las zonas junto con detección de temperatura para poder analizar los cambios de esta y detectar emisiones sin poner en peligro a una persona.

También son utilizados para la inspección y vigilancia, incluso reparación de ciertas partes de infraestructuras de difícil acceso como puede ser torres de alta tensión o parques eólicos junto con los molinos («DJI: “La industria del dron crece gracias al uso profesional”» 2017).

RECREATIVO

Quizá uno de los mayores usos actuales es el recreativos, ya que el aumento de los drones que pequeño tamaño y ofrecidos como juguete permite a muchos aficionados realizar sus propios vuelos y grabaciones. Al mismo tiempo el fácil acceso a estos dispositivos junto con el desconocimiento de la normativa y manejo de las aeronaves también es uno de los sectores que más riesgo conlleva para el vuelo seguro.

EVOLUCION DE LOS DRONES

Como hemos visto anteriormente se distinguen diferentes sectores en los que se emplean estos dispositivos, a continuación, diferenciaremos únicamente los usos recreativos, militares y los usos empresariales en los que recogeremos el resto de los usos.

Desde el desarrollo de los drones tal y como hoy los conocemos el uso e interés ha ido aumentando, en 2017 en ciclo de sobre expectativa aparecieron los drones como una de las tecnologías emergentes (ver Figura 2) y de las cuales suscitaba interés tanto para usuarios como para inversores.

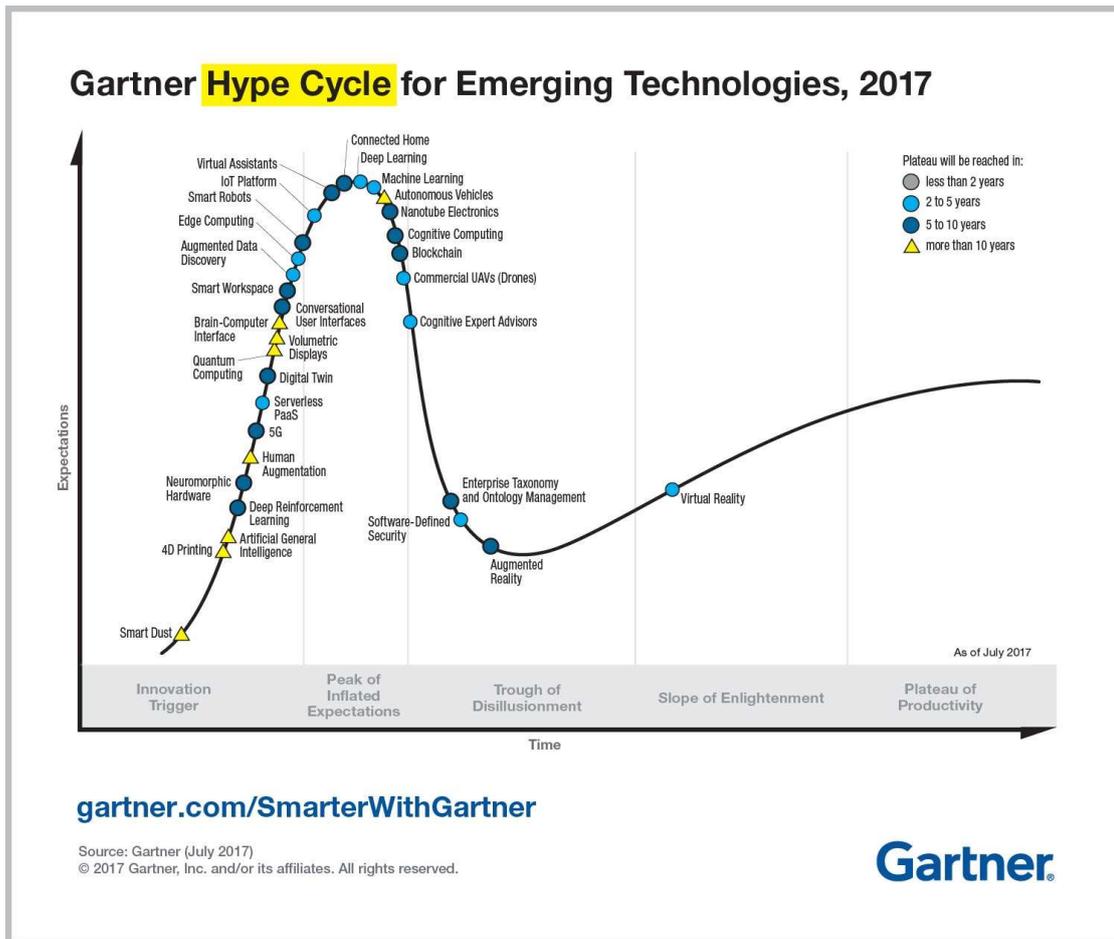


Figura 2: Ciclo de sobre expectativa de Gartner 2017 («Top Trends in the Gartner Hype Cycle for Emerging Technologies, 2017» s. f.)

En este caso se encuentra en la fase conocida como “abismo de desilusión”, en la que el interés de inversores se ve reducido, no es una tecnología suficientemente innovadora o no cumplen las expectativas creadas. Esto puede llevarnos a pensar que es una tecnología poco útil o que no se ha llevado al mercado, pero no es así ya que tras esta fase nos encontramos con la rampa de consolidación en la que estos dispositivos siguen generando cierto interés lo que llevara a desarrollarlos hasta llegar a la meseta de productividad en la que los beneficios quedan ampliamente demostrados. Actualmente los drones podrían encontrarse en esta etapa final, en la que se conocen los beneficios y empieza a implementarse su uso.

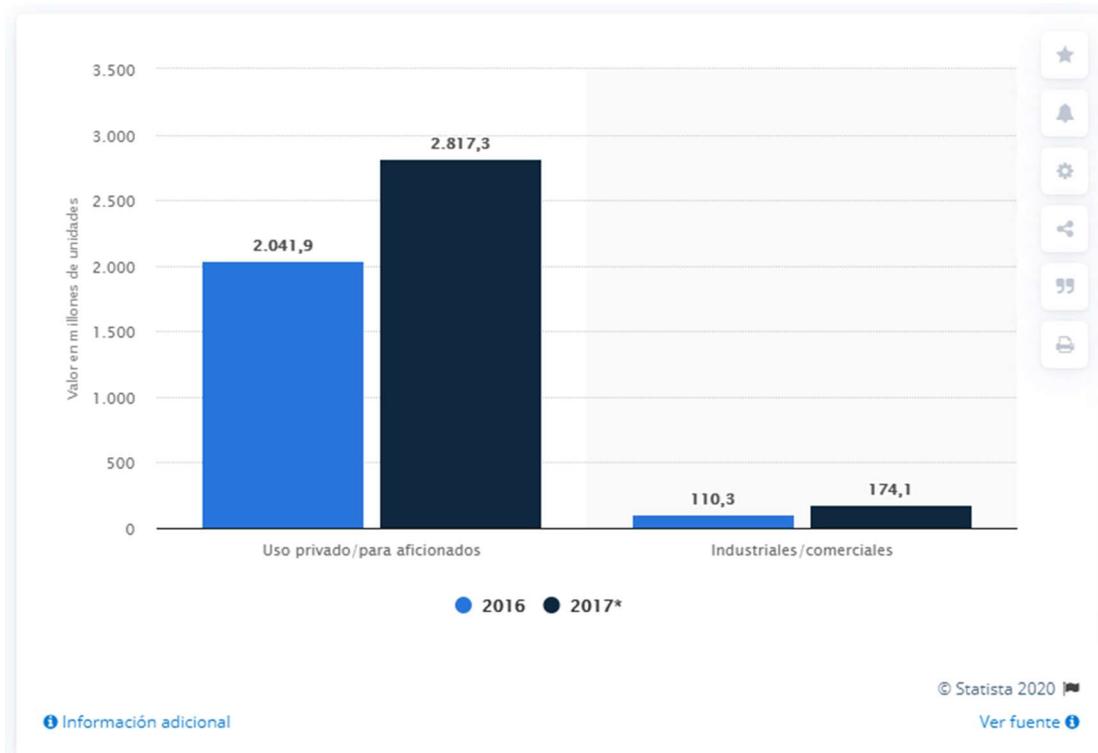


Figura 3: Venta de drones en 2016 y 2017 (Rosa Fernández 2019)

Como podemos ver en la Figura 3 que muestra las estadísticas de número de ventas de aeronaves no tripuladas, esta estadística desarrollada por “Statista” podemos ver el aumento significativo de ventas de aproximadamente 800 mil aeronaves entre 2016 y 2017 en el sector privado, siendo menos notable en el ámbito comercial. Esto puede deberse como hemos visto con anterioridad a que durante 2017 no cumple las expectativas o no se encuentra gran utilidad.

Con los siguientes años podemos ver que la cifra aumenta y también lo comienza a hacer la parte comercial de la venta de drones, debido a comienzan a darse uso a estos dispositivos en diferentes actividades comerciales y por otra parte el coste de estos se ve reducido, lo que supone un gran impulso al desarrollo.

Uno de los usos más destacados y en torno a 2017 y que potencio el uso del dron de forma comercial fue en el sector topográfico y de inspección. Según un estudio desarrollado por “Drone Industry Insights” el 80% de los drones de uso comercial se encontraban en este sector. Este hecho da lugar a abrir la puerta a estos dispositivos a otros sectores en los que ocuparan un lugar importante.

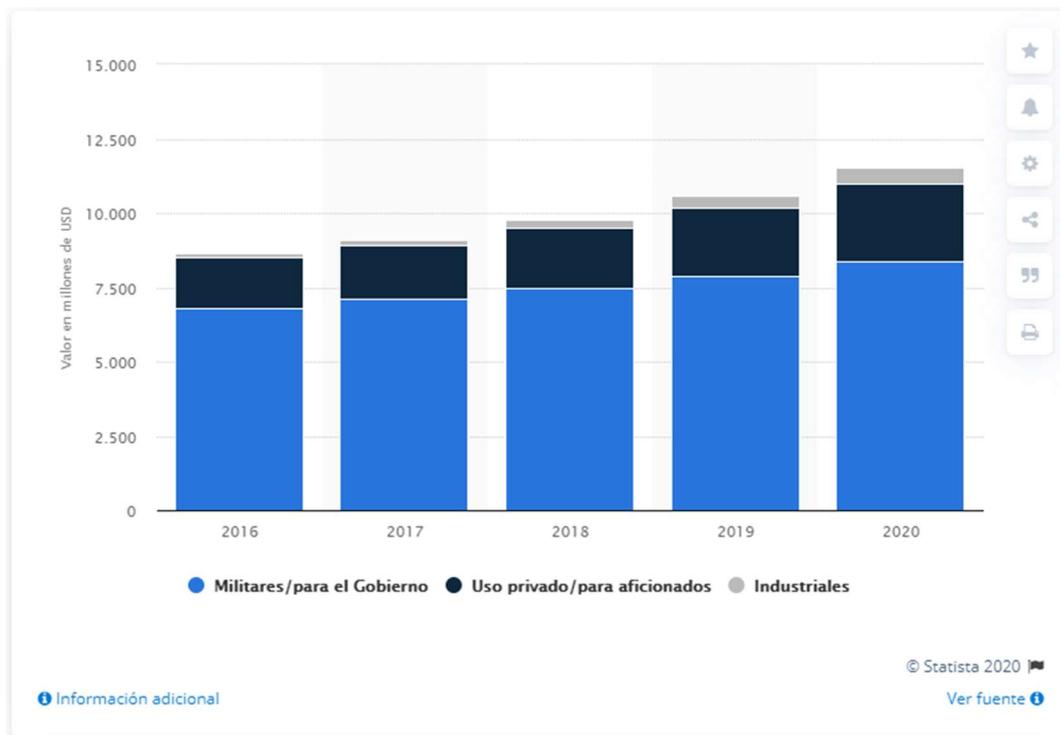


Figura 4: Valor económico del mercado de drones (Rosa Fernández 2019)

En la Figura 4 podemos ver una estimación del valor del valor esperado en millones de dólares procedente de la venta de drones a nivel global. Podemos ver los tres usos principales: militares, recreativos e industriales. Como podemos observar el valor total ha aumentado respecto al año anterior, a continuación, analizaremos los datos de forma individual. En primer lugar, observamos que el ámbito militar es el que mayor valor aporta, también hay que tener en cuenta que se produce un aumento cada año, pero no tan pronunciado como los otros dos casos. En cuanto al uso recreativo vemos que los primeros años aumenta, pero sin destacar, en cambio en 2018, 2019 y 2020 el aumento es mayor. Por último, analizando el sector industrial vemos que es el que menor valor aporta, esto puede deberse a no encontrar una actividad a la que adecuar este tipo de dispositivos. En cambio, en se espera que casi duplique el valor respecto al 2019, pasando de 370 millones a 530 millones. Este aumento se debe a que ha comenzado a ponerse en práctica y demostrar los beneficios en muchos sectores, como puede ser agricultura, incendios forestales o incluso transporte de mercancía.

Tras analizar estos datos podemos afirmar que el sector que supone mayor gasto en el sector de las aeronaves no tripuladas es el militar, como suele ser habitual es el sector que busca desarrollar e incorporar la tecnología más innovadora.

Por otra parte, vemos la tendencia del sector industrial a aumentar la inversión en este tipo de dispositivos después de comprobar el potencial y las ventajas de contar con estas aeronaves como parte del equipo.

Por último, aunque no menos importante, analizando el sector recreativo nos damos cuenta de que es una parte importante del valor mundial de 2300 millones en el año 2019. Esto nos da una aproximación de la cantidad de drones para uso particular, por distintos tipos de personas, en muchos de los casos llegando a ser niños y también con escasos conocimientos de reglamentos y vuelo de estas aeronaves, lo cual puede suponer un problema en varios ámbitos como veremos más adelante.

DRONES EN EL SECTOR INDUSTRIAL

INCENDIOS FORESTALES

En España la empresa multinacional Telefónica comenzó un proyecto en el que dotar de drones a los cuerpos de bomberos y servicios de emergencia para detectar y controlar rápidamente un posible incendio. Para ello empleando las torres de telecomunicaciones para instalar sensores de térmicos que permitan enviar señales a los drones para que estos actúen rápidamente y recoger toda la información en tiempo real para enviarla a los servicios de emergencia para que puedan actuar en consecuencia.

Al mismo tiempo se encuentra trabajando en el desarrollo de un dron capaz de sofocar pequeños focos de forma autónoma.

AGRICULTURA

Actualmente en Córdoba se ha llevado a cabo un ejemplo de uso de los drones en la agricultura. En este caso tiene lugar en los olivares, en los que se emplea un dron para georreferenciar todos los cultivos y de esta forma llevar a cabo un análisis de todos los datos recogidos por el dispositivo. Este análisis permitirá aumentar el rendimiento del cultivo y al mismo tiempo aumentar el beneficio ya que habrá un

mayor de las diferentes partes de la plantación y por tanto mejora la forma de actuar en distintas circunstancias (Córdoba 2020).

CONSTRUCCIÓN Y CARTOGRAFÍA

Un ejemplo reciente de la importancia de esta tecnología en este sector es, la decisión del Departamento de Cohesión Territorial del Gobierno de Navarra que ha comenzado a utilizar drones para obras públicas. De esta forma mediante los dos drones pueden realizar seguimiento y planificar nuevos proyectos de construcción de una forma más eficiente y segura. Una vez realizadas las obras, se emplearán estos dispositivos para tomar fotografías y actualizar la información cartográfica referente a carreteras.(Europa Press 2020)

VIGILANCIA EN CARRETERAS

Actualmente en España un sector en el que se emplea las aeronaves no tripuladas es en la Dirección General de Tráfico para el control del tráfico en distintas vías. Para ello utilizando un modelo de la marca DJI equipado con un cámara de gran calidad lo que permite grabar las posibles sanciones de los conductores como el no uso del cinturón de seguridad, lanzamiento de colillas... A pesar de esto no puede ser empleado para controlar el exceso de velocidad debido a limitaciones tanto de la tecnología como de la legislación. Por otra parte, entran en juego otras dos limitaciones que son la privacidad, que imágenes son grabadas, donde se almacenan dichas imágenes... La otra limitación tiene que ver con la normativa de vuelo de estas aeronaves ya que no se puede sobrepasar los 120 metro de altura y solo pueden actuar en un radio de 500 metros. Por estos motivos no ha tenido gran éxito dicho uso (Villareal 2019).

PREVENCIÓN DE CONTAGIO

Durante el inicio del año 2020 ha surgido y proliferado un nuevo virus conocido como Coronavirus, debido al desconocimiento de este y el tratamiento a realizar junto con la rápida proliferación de este a llevado a las Autoridades Chinas a utilizar drones como forma de control de los ciudadanos. Los drones empleados contaban con una grabación en la que se daba indicaciones a los ciudadanos de cómo actuar y

las acciones recomendadas ante tal fenómeno, incluso siendo utilizado por las autoridades para dar mensajes directos a las personas como podía ser indicar que utilizaran la mascarilla (Granda 2020).

Tras la llegada de dicho virus a España también se han implantado medidas de contención lo que ha dado cabida al uso de drones para controlar el cumplimiento de la distancia de seguridad y la buena conducta de los ciudadanos. El fin de semana del 8 de junio la Policía Local de Lorca empleo drones para el control de la playa mediante la visualización de las imágenes facilitadas por los dispositivos.(La actualidad 2020)

BÚSQUEDAS Y RESCATES

Actualmente Suecia está realizando pruebas para el transporte de equipos desfibriladores en los casos que se sospeche que una incidencia pueda estar causada por un paro cardiaco. Esta investigación está siendo llevada a cabo por SOS Alarm, el Centro de Ciencia de Reanimación del Instituto Karolinska y la compañía de software Everdrone. De esta forma se espera que pueda utilizarles las tecnologías incluidas en un dron como son el GPS y la toma de imágenes para llevar el desfibrilador a cualquier parte del territorio en el menor tiempo posible intentando salvar la vida del paciente. (Bullock 2020)

Otro caso, aunque esta vez más anecdótico, en la provincia de Valladolid tras el avistamiento de un posible cocodrilo en la zona donde confluyen los ríos Duero y Pisuerga en el que los equipos de vigilancia emplean un dron para facilitar la búsqueda. (EDCM/EP 2020)

FUTURO DE LOS DRONES

Hemos visto cómo se encuentra el sector de las aeronaves no tripuladas en la actualidad, a continuación, veremos las previsiones para el futuro.

Según un estudio realizado por la *International Data Corporation* la inversión mundial realizada en el ámbito de los drones aumentará en 2020 en torno a un 34% respecto al año anterior, lo que supone un total de 16300 millones de dólares.

La estimación del gasto por parte de los consumidores de este sector es de 6500 millones de dólares, la mayor parte de este gasto está relacionado con gobiernos y/o instituciones educativas.

Otro dato interesante es que los mayores beneficiados de este aumento tanto de la inversión como del gasto son: China y Japón, seguidos por EE. UU. y por Europa Occidental. Esto se debe a que la mayor parte de las marcas de drones proceden de Oriente al igual que las empresas relacionadas con la fabricación de componentes. Respecto al sector industrial el número de ventas estimadas por *Gartner* es de aproximadamente 526000 unidades la mayoría de ellas empleadas para la monitorización de la construcción, estimando que para 2023 las unidades vendidas sean 1.3 millones. Estos datos demuestran que es una tecnología en auge y que el aumento de los mismo conlleva grandes retos que afrontar y oportunidades que aparecerán (Valdeolmillos 2020).

Otro sector en el que se prevé que aumente el uso de drones es en la industria de los seguros, llegando a tener 136000 dispositivos en esta actividad. Estas actividades estarán basadas en la inspección y vigilancia de edificios o estructuras en las que se haya hecho una reclamación para analizar el alcance y las causas de los daños. También podrá estudiar la estructura de un edificio para realizar una cotización de seguro de este, todo ello como en la mayoría de los casos de uso de drones, reduciendo los costes y evitando situación de peligro a las personas, en este caso sin necesidad de montar andamios o estructuras para su análisis («Las ventas de drones empresariales crecerán un 50% en 2020» 2019).

Este aumento de los drones y la previsión de que siga aumentando unido a otros factores que veremos después han dado lugar a que las autoridades comiencen a controlar el uso de estos. En España en febrero de 2020 se implantaron 54 equipos PEGASO (Policía Especialista en Gestión Aeronáutica y de Seguridad Operacional) para la vigilancia del espacio aéreo, se trata de un servicio de la Guardia Civil para el control de drones. Se centrarán en el control de zonas próximas a aeropuertos, helipuertos o pistas de emergencia para combatir el mal uso de drones dispondrán de Sistema Global Contra Dron (SIGLO-CD) para detectar el vuelo de drones.

Otra de las actividades realizadas por este grupo será realizar inspecciones a escuelas de vuelo, servicios profesionales y también impartir actividades de concienciación orientadas a usuarios de drones. Para evitar el vuelo en zonas no permitidas o que no son seguras para tal actividad disponen de un bloqueador de señal que interrumpe la comunicación entre el controlador y la aeronave obligando al dispositivo a realizar un aterrizaje y finalizar el vuelo («La Guardia Civil despliega la “policía del aire” para controlar el vuelo ilegal de drones» 2020).

Empresas como Amazon, UPS y otras agencias de transporte o reparto están investigando y realizando intentos de envíos de paquetería mediante drones autónomos. Este tipo de envíos suponen algunas ventajas sobre el transporte tradicional como el acceso a zonas de difícil acceso, reducir la contaminación debido al no uso de combustibles fósiles, reducción de los costes... No todo son ventajas ya que también se encuentran muchas limitaciones, la principal limitación está determinada por la normativa de vuelo de aeronaves no tripuladas. La normativa actual es bastante restrictiva con el uso de este tipo de dispositivos, algunas limitaciones para este tipo de uso son: la necesidad de un piloto del dron que mantenga visión directa con el dispositivo, restricción de vuelos en muchas zonas como proximidades de aeropuertos, núcleos urbanos sobrevolando la población... Otras limitaciones están condicionadas a la estructura del dron como puede ser el peso máximo que puede transportar en la mayoría de los casos este peso es bastante reducido por lo que se limita al envío de paquetes pequeños (Aguiar 2020).

En diciembre de 2016 Amazon realizó el primer envío de un pedido a través de un dron, este primer envío se realizó en Reino Unido siendo un envío de prueba bajo condiciones controladas. A pesar de estos problemas, las empresas siguen con sus investigaciones e intentos por conseguir realizar los envíos reales a través de estas aeronaves («La entrega con drones de Amazon es real: así fue el primer envío a un cliente» 2016).

PROBLEMAS RELACIONADOS CON EL USO DE DRONES

En la actualidad nos encontramos con bastantes problemas relacionados con el mal uso de los drones, esto tiene lugar principalmente en el ámbito recreativo debido al desconocimiento de lo que supone la utilización de una aeronave no tripulada, a pesar de que sea tratado como un juguete en algunos casos. El otro factor que da lugar a un uso inadecuado de estos dispositivos es el desconocimiento de la normativa, que en el caso de España se recoge en el Real Decreto 1036/2017, en el que se recogen todas las medidas, restricciones y recomendaciones para el uso de aeronaves no tripuladas tanto para uso recreativo como para uso profesional, indicando en tal caso que requisitos tiene que cumplir el piloto y las licencias y permisos necesarios en los diferentes casos.

Además del desconocimiento de la normativa, también existe el desconocimiento y la gravedad del incumplimiento del reglamento lo que puede llevar a distintas sanciones algunas de ellas graves. Por esto motivo surge la idea de desarrollo de una aplicación que permita acercar a los usuarios la normativa y evitar comportamientos inadecuados con drones que pongan en riesgo al resto de la población.

A pesar de que el aumento de personas que disponen de un dron y el intento de concienciación por parte de las autoridades de un uso responsable del mismo, con medidas como la creación del grupo PEGASO como hemos señalado anteriormente, todavía nos encontramos acciones que incumplen la normativa.

El 15 de febrero de 2020, un usuario fue denunciado por los Mossos d'Escuadra por realizar un vuelo sobre la autovía A-14 en una ubicación en la que la distancia al aeropuerto era de 1.5 km aproximadamente, encontrándose en el interior de una zona restringida al vuelo ya que está dentro del radio de 8km tomando como centro el aeropuerto donde se prohíbe el vuelo. Este tipo de acciones pueden suponer multas de hasta 225 mil euros en casos de que el piloto no sea profesional y de hasta 4.5 millones de euros si es profesional (Gabriel 2020).

El 6 de marzo de 2020, otro usuario fue denunciado por grabar un evento con un dron de 500gr, este usuario incumplía distintas normas como son volar dentro del

alcance del aeropuerto, volar sobre una multitud, incluso podría suponer el incumplimiento de la ley de privacidad en caso de que no se disponga de permiso de los asistentes para la grabación. Estas acciones pueden ser penalizadas incluso si se tratase de un dron de 250gr o menos, considerados de juguete («Denunciado por volar en Elche un dron para grabar un evento» 2020).

Un caso que ha tenido gran repercusión tuvo lugar en Madrid el 8 de febrero de 2020, en el que la presencia de un objeto que los pilotos y controladores identificaron como un dron, llevo a paralizar el tránsito aéreo durante unas horas. Esta acción supuso el desvío de 26 vuelos y miles de pasajeros afectados. A pesar de la gravedad de la situación, este tipo de acciones pueden causar daños aún mayores, por lo que es necesario tomar medidas para que no se produzcan (Salvador 2020).

Otro caso similar al mencionado anteriormente, sin tener tantas consecuencias como cerrar el aeropuerto debido a la baja actividad tuvo lugar el 26 de mayo de 2020. En cuestión una persona sobrevolaba un dron en el aeropuerto de Vigo en la entrada de la pista de aterrizaje, lo que supuso una denuncia por parte de la Policía Nacional. (Pita 2020)

Por otra parte, según datos de la Agencia Estatal de la Seguridad Aérea (AESA), el número de incidentes relacionados con el vuelo de drones en Cataluña ha pasado de los 10 casos en 2015 a 54 en 2019, siendo el total de casos en España de 470 en 2019. Algunas de estas incidencias se han producido en ciudades como Barcelona o Madrid que también forman parte del espacio aéreo controlado (Rocasalva s. f.).

Estos solo son algunos ejemplos que nos podemos encontrar de un mal uso de los drones, pero no los únicos, esto es debido al gran desconocimiento de la normativa. Por lo que es importante hacer llegar esta información junto con recomendaciones de vuelo a los usuarios, sobre todo a usuarios cuyo uso es el recreativo.

Esta situación nos lleva al desarrollo de la aplicación que pueda hacer llegar la normativa a los usuarios de un forma más cómoda e intuitiva de manera que se reduzca este tipo de situaciones.

EASYDRON

EasyDron se trata de una aplicación web adaptable a todo tipo de dispositivos que permite obtener información relacionada con la posibilidad de realizar vuelos con dron en una determinada zona. El nombre elegido hace referencia a la filosofía de la aplicación, Easy cuya traducción al castellano es “fácil”, y dron haciendo referencia al dispositivo. El objetivo es presentar una aplicación fácil de utilizar al usuario y que ofrezca la información con simpleza para que el usuario no pierda mucho tiempo en aprender a manejar la aplicación y resulte atractivo su uso.

OBJETIVOS

A pesar de que en la actualidad muchos de los drones del mercado incorporan la tecnología No Fly Zone, que consiste en impedir el vuelo de los drones en zonas restringidas (aeropuertos). Esta tecnología la incorporan los fabricantes introduciendo un mapa de los aeropuertos ya que son zonas de vuelo prohibido de forma que el dron cuando detecte que se encuentra en una de esas ubicaciones realice un aterrizaje o vuelva al punto de partida.

Además de analizar estas zonas también resulta de gran interés contemplar otras zonas, ya que como hemos visto en el apartado anterior no solo hay incidentes en las proximidades de aeropuerto. En España nos encontramos también con zonas de especial protección de aves o parques naturales en los que existe restricciones de vuelos para preservar la fauna y flora sensible, como podemos ver en la disposición adicional undécima de la Ley 33/2015. Por otra parte también existe la limitación del vuelo en zonas urbanas, aglomeraciones de edificios o reuniones de personas como se indica en el Capítulo III del Real Decreto 1036/2017 («BOE-A-2017-15721.pdf» 2017.)

El objetivo principal de la aplicación es permitir al usuario comprobar si el lugar o zona tanto de manera genérica como personalizada por donde planea realizar un

vuelo es una zona adecuada, en la que no se hay ningún tipo limitación teórica contemplada por la normativa como puede ser el vuelo en las proximidades de un aeropuerto. De esta forma el usuario podrá indicar la zona ya sea únicamente un punto o un trazo de ruta que ocupe distintas zonas y mediante el uso de bases de datos espaciales realizar los cálculos pertinentes para obtener el resultado de la consulta. Al utilizar el dibujo de rutas no solo se limita a un punto y se evita situaciones como una ubicación que se encuentre al límite una zona con restricción al vuelo, en ese punto se podrá volar, pero en las proximidades en alguna dirección no, de esta forma se podrá encontrar una zona correcta para el vuelo final.

A demás también se acercará la normativa al usuario mediante consejos, ya que puede existir el caso en el que la posición geográfica permite el vuelo de aeronaves no tripuladas, en cambio la situación en ese instante puede que no lo permita, por ejemplo, en el caso de que haya una multitud de personas por lo que el vuelo no estaría permitido. Para ello, la aplicación dispondrá de un apartado en el que se muestre una serie de consejos formulados a partir de la normativa de vuelo, pero de una forma más sencilla de comprender.

Otro de los puntos que cubre la aplicación es dar información sobre la autoridad que gestiona la zona de vuelo para poder contactar en caso de que sea necesario solicitar un permiso para la realización de un vuelo, ya sea por uso profesional o recreativo. Al mismo tiempo esta información permite al usuario contactar con la autoridad responsable para comunicar cualquier tipo de incidencia que surja en la zona. Como se ha mencionado anteriormente no se tomará únicamente restricciones basadas en la proximidad a zonas de vuelo, bien sean aeropuertos, helipuertos, bases aéreas militares o aeródromos, también se tendrá en cuenta zonas de protección de aves en las que el vuelo puede estar limitado dependiendo de la autoridad que lo gestione y otros factores. Los núcleos urbanos como las ciudades en las que el vuelo está limitado bajo ciertas condiciones como pueden ser peso del dron, altura de vuelo, permisos de vuelo... También se tendrá en cuenta las zonas de parques naturales ya que pueden presentar restricciones al vuelo por lo que será necesario ponerse en contacto con la autoridad para consultar la posibilidad y/o solicitar el permiso pertinente.

ESTUDIO DE MERCADO

En la actualidad existen aplicaciones que realizan valoraciones de las rutas o zonas de vuelo para un dron. Además, aportan información al usuario del dron que facilitan realizar un vuelo con seguridad. Algunas de ellas son:

RUNAWAY HD

Es necesario realizar un registro para utilizar esta aplicación que está disponible únicamente en sistemas iOS. Esta aplicación nos proporciona información de que zonas próximas al piloto se encuentran restringidas al vuelo.

UAV FORECAST

Se trata de una aplicación disponible tanto para iOS como para Android que muestra información relativa a zonas restringidas al vuelo, condiciones meteorológicas y actividad solar. Incluye en la pantalla de inicio un resumen de las condiciones de vuelo basadas en las condiciones climáticas, no tiene en cuenta la posición geográfica a la hora de dar esta información.

Por otra parte, no se muestra de forma clara la restricción al vuelo, no es visible un mensaje de advertencia. Respecto a la información meteorológica es muy completo ya que indica tanto pronóstico, tiempo actual, velocidad y dirección del viento. En relación con la ubicación únicamente toma como posición un punto a partir del que verifica la posibilidad de vuelo. Al mismo tiempo sobre el mapa solo se muestran restricciones debido a zonas relacionadas con aeropuertos. Es una aplicación muy completa en la información climática, pero en cuanto a la información de restricciones basada en la ubicación puede ser mejor.

KITTY HAWK

Esta aplicación difiere del objetivo principal por el cual se desarrolla la aplicación, pero resulta interesante su uso. El objetivo de KITTY HAWK es el almacenamiento de datos relativos a los vuelos realizados: rutas, tiempo de vuelo, meteorología... lo que permite al usuario tener al alcance un resumen de sus vuelos.

DRONE BUDDY

Drone Buddy al igual que el resto de las aplicaciones muestra las zonas en las que el vuelo está restringido debido a la proximidad de un aeropuerto o aeródromo junto con información meteorológica y velocidad del viento. Respecto a la información geográfica igual que en el caso de UAV FORESCAST es limitada ya que se podrían tener en cuenta otros factores como zonas de protección de aves o advertencia de proximidad a núcleos urbanos. En este caso tanto la información como el uso resulta menos intuitivo que en la aplicación mencionada anteriormente. En el aspecto meteorológico también nos encontramos con menos parámetros en este caso únicamente con temperatura, posición del sol y velocidad del viento. Podemos encontrar esta aplicación en iOS o Android.

ENAIRE

La aplicación desarrollada por el gestor de la navegación aérea española ENAIRE, en primer lugar, ofrece la posibilidad de visualizar un mapa específico para los vuelos recreativos o bien un mapa de carácter profesional. Una vez seleccionada la opción deseada podemos ver un mapa que nos ofrece la posibilidad de seleccionar diferentes capas para visualizar como: zonas de especial protección para aves, espacios naturales protegidos, aeródromos... A la hora de obtener información de una zona concreta permite la realización tanto de una ruta con diferentes formatos de dibujo: líneas, punto, polígono... A pesar de las diferentes opciones ofrecidas para el dibujo, su uso no resulta muy intuitivo lo que puede generar dificultades, sobre todo en primeros usos. Esto se debe a que la forma en la que seleccionar el tipo de dibujo para la ruta no resulta muy claro, además a la hora de realizar el dibujo sobre la zona deseada no resulta muy intuitivo.

Por otra parte, a la hora de mostrar un resultado no resulta fácil de visualizar ya que no se dispone de un botón en el que validar y/o visualizar los resultados.

A pesar de los inconvenientes de la aplicación debidos a la poca amigabilidad de su uso es una aplicación muy completa respecto a información que utiliza para validar una ruta, además debido a que se trata de una aplicación de una autorización responsable de los vuelos tiene mayor fiabilidad.

GEO ZONE MAP

DJI, uno de los mayores fabricantes y vendedores de drones dispone de una página web en la que se puede visualizar sobre cartografía las zonas que presentan restricciones de vuelo ya sean por pertenecer a un aeropuerto o ser zonas de ocupación militar.

No aporta más información que la visualizada en el mapa, no se puede introducir tu ubicación, ni obtiene la ubicación en la que te encuentras, lo que no proporciona un uso fluido y en tiempo real. Una de las características a tener en cuenta de esta aplicación es el filtro por los distintos modelos de drones para mostrar las restricciones de vuelo.

AIRMAP

Se trata de una aplicación en la que a través de la ubicación única de un punto permite la evaluación de un vuelo, en este caso se incluye restricciones de aeropuertos, aéreas de uso aéreo especial que no se identifican de forma clara, que se indican mediante avisos informando de posibles peligros, mediante avisos NOTAM. Esta aplicación también incluye un apartado de los vuelos planificados y la posibilidad de registrar el dron, para ello es necesario estar registrado en la aplicación. En caso de estar registrado puedes realizar el dibujo de la ruta sobre el mapa de diferentes formas una vez dibujado obtienes los resultados al respecto sobre la planificación.

La información que muestra como resultado son las condiciones meteorológicas, a que autoridades deniegan el vuelo y una serie de recomendaciones de vuelo. En caso de que la planificación no sea favorable se indica las reglas que se están incumpliendo. En cuanto al registro de la aeronave permite incluir el nombre y el modelo utilizado. Esta aplicación la podemos encontrar tanto en aplicación web como aplicación móvil en Android y iOS. Esta es un ejemplo de las aplicaciones más completas con las que nos podemos encontrar debido a la información mostrada y las diferentes posibilidades que nos ofrece.

Estas son solo un ejemplo de las múltiples aplicaciones para evaluar el vuelo de un dron en una localización concreta. En conjunto nos podemos encontrar con algunas

limitaciones en estas aplicaciones como puede ser que no permitan dibujar el recorrido por el que se quiere volar, aunque alguna como AirMap si lo permiten.

La mayoría de estas aplicaciones únicamente presenta información relacionada con aeropuertos, helipuertos o aeródromos sin tener en cuenta zonas donde hay limitación al vuelo como pueden ser núcleos urbanos o zonas de protección de aves o parques naturales. Además de esta limitación en cuanto a la información de restricciones también nos encontramos con que la representación y explicación de esta información no es clara pudiendo ser un problema para un usuario inexperto. Por otra parte, la información que se muestra en muchos casos puede ir destinada a un público con experiencia en el vuelo de drones ya que se incluye bastante información meteorológica y notaciones NOTAM que pueden ser desconocidas para gran parte de los usuarios. Además, también estas aplicaciones tienden a no tener en cuenta información respecto a zonas como poblaciones en las que también existen ciertas limitaciones al vuelo, pudiendo causar desinformación a un usuario con gran desconocimiento de las normas de vuelo de estos dispositivos.

Por otra parte, la mayoría de estas no incluye la posibilidad de registrar un dron y ajustar la evaluación y/o consejos a las características de este, lo que resulta de interés ya que las limitaciones al vuelo varían también en función del tipo y peso del dron.

Otro dato que podemos obtener de este estudio es la cantidad de descargas y por tanto interés y/o uso de estas aplicaciones superando algunos casos como AirMap las 100.000 descargas en dispositivos Android sin tener en cuenta las descargas en sistemas iOS ni el uso mediante aplicación web. Estos datos nos dan una idea de la cantidad de personas que realizan un uso de estas aplicaciones en sus diferentes estilos y formatos para obtener información acerca de las condiciones de vuelo. Esto unido con el crecimiento del mercado de los drones y el fácil acceso de los mismo supone una posibilidad de desarrollo y mejora de las aplicaciones relacionadas con la seguridad de vuelo para drones.

ANÁLISIS

Como se ha visto en el apartado “Problemas relacionados con el uso de drones”, la multitud de incidentes y problemas causados por los drones han ido aumentando al igual que lo ha hecho el mercado de estos. Por ello es necesario tratar de reducir este tipo de problemas por lo que en primera instancia se crea una normativa, en el caso de España el Real Decreto 1036/2017. A pesar de esto muchas de las personas que acceden a uno de estos dispositivos desconoce la normativa, un ejemplo de esto es que la mayor parte de los incidentes tienen lugar en las proximidades de los aeropuertos siendo unos de los puntos más importantes y sobre los que mayor énfasis se hace en el real decreto.

Para facilitar un buen uso de estos dispositivos puede hacerse mediante el desarrollo de aplicaciones que faciliten el acceso y cumplimiento de la información recogida en la normativa.

En primer lugar, es necesario que los usuarios de los drones conozcan que zonas son seguras para realizar un vuelo con su dispositivo o por lo contrario si se trata de zonas en la que existen restricciones a este tipo de actividad. Como se ha mencionado anteriormente en España existen cuatro zonas principales que pueden presentar restricciones de vuelo de aeronaves no tripuladas, que son: Zonas de Especial Protección de Aves (ZEPAs), Parques Naturales, Poblaciones y Aeropuertos. Por lo tanto, esta será la principal fuente de información para obtener una zona de vuelo segura con un dron.

A pesar de que las mayores limitaciones tienen lugar en las zonas de aeropuertos debido a la gravedad de las consecuencias que puede causar tanto en temas de seguridad pudiendo provocar accidentes aéreos tanto económicos debidos al cierre de un aeropuerto como se ha producido en algunos casos mencionados en el apartado anterior, también es necesario dar a conocer otras zonas para que posibles usuarios inexpertos tengan a su alcance información que pueden desconocer.

En el caso de los aeropuertos existe una gran limitación que no solo se reduce a la zona de este, sino que también lo hace a un radio de 8km ya que forma parte de zonas en las que el tráfico aéreo es alto y crítico ya que se realizan las aproximaciones para aterrizajes y despegues.

Por otra parte, en las Zonas de Especial Protección de Aves y Parques Naturales puede estar restringido el vuelo para evitar causar daños y perturbar tanto a la fauna como a la flora del lugar.

Por último, está la limitación al vuelo en poblaciones, en este caso las limitaciones se deben a la concentración de edificios, aglomeraciones de personas... Esta zona presenta limitaciones, pero con peculiaridades ya existen situaciones en las que está permitido el vuelo bajo ciertas condiciones. Por ejemplo, en caso de que el dispositivo tenga un peso menor de 250 gramos pueden realizarse vuelo siempre y cuando sea una zona despejada y no se vuele sobre personas.

Para solventar este primer problema relacionado con las zonas de vuelos será necesario evaluar la posible ruta seleccionada por el usuario mediante el dibujo de esta. A partir de la geometría de la ruta dibujada por el usuario se comprobará si existe alguna zona con restricción al vuelo en la zona que ocupa la ruta.

Estas zonas se tratarán en función del grado de limitación: En las poblaciones la limitación puede ser menor en función del lugar, del peso del dron y otros factores. Un grado medio de limitación en el caso de parques naturales y ZEPAs por lo que se trataran los datos de ambos de forma conjunta y, por último, un alto grado de prohibición al vuelo en el caso de los aeropuertos.

En el caso de parques naturales y ZEPAs será necesario diferenciar varios casos:

1. Caso 1: Existe tanto parques naturales como ZEPAs en el encuadre de la ruta. Por lo que se trataran los datos de forma conjunta a través de la unión de las dos capas.
2. Caso 2: Solo existen parques naturales en el encuadre de la ruta. Únicamente se tendrá en cuenta la capa de parques naturales para el procesamiento.
3. Caso 3: Solo existen ZEPAs en el encuadre de la ruta. Únicamente se tendrá en cuenta la capa de ZEPAs para el procesamiento.
4. Caso 4: No existe ninguna de estas zonas en el encuadre por lo que no se realizará ningún procesamiento.

Tras la comprobación de la validez de una ruta se pretende mostrar al usuario gráficamente las zonas en las que no se permita el vuelo, no solo mediante información textual como se indica en la mayoría de las aplicaciones del mercado. Esta mejora permitirá al usuario visualizar más fácilmente que zonas de la posible ruta presentan limitaciones.

Otro de los factores a tratar es la amigabilidad y facilidad de un usuario inexperto con este tipo de aplicaciones para obtener la información de forma clara y concisa. Ya que como se ha observado y comentado en el apartado de estudio de mercado, tras analizar varias aplicaciones en el mercado muchas de ellas pueden resultar difíciles de utilizar y de comprender en el instante de uso.

Por ello desarrollar una aplicación que sea intuitiva a la hora de que el usuario pueda planificar una ruta, es decir, que pueda incluir o dibujar la ruta que desea valorar. Para ello, se emplearán botones que sean suficientemente aclarativos mediante el uso tanto de texto descriptivo como de iconos.

No solo se pretende conseguir facilidad en el uso de la aplicación en función de su interfaz, sino que sea intuitivo a la hora de llevar a cabo los pasos de planificación. De esta forma el usuario tendrá mayor facilidad para saber en cada momento cual es el siguiente paso que necesita dar para planificar el vuelo y obtener la validación correspondiente.

Otro de los apartados que debe tenerse en cuenta es la información presentada al usuario, de forma que se incluya toda la información imprescindible para que el usuario tenga conocimiento en todo momento del resultado obtenido tras la validación de la ruta y no sobrecargar al usuario de información que pueda no ser muy útil y pueda causar confusión en este.

Por otra parte, debido al aumento de drones en el ámbito recreativo del que forman parte la mayoría de los incidentes causados como se ha visto anteriormente, se pretende concienciar a los nuevos usuarios de la importancia de respetar las normas de vuelo para estos dispositivos. Por este motivo es importante dar a conocer tanto las normativas vigentes como las recomendaciones de las autoridades a la hora de

realizar un vuelo con un dron. Este apartado puede cubrirse mediante consejos de vuelo obtenidos mediante una recopilación de la información mencionada anteriormente para permitir que el usuario llegue a ella fácilmente, además de darle a conocer las autoridades competentes y la información original de los consejos para que ellos mismos puedan tomarlas como fuente principal ante cualquier duda.

DISEÑO

A continuación, se detallará las soluciones y desarrollos para llevar a cabo la aplicación haciendo frente a los problemas mencionados anteriormente.

INTERFAZ

En este apartado se tratará el problema de la amigabilidad de la aplicación en lo referente a presentación y muestra de datos. Para conseguir este objetivo es necesario tener en cuenta varios componentes de la aplicación, como son: estructura, información, tipos de letra, tamaños de letra, diseño de la interfaz.... Estos componentes se realizarán teniendo en cuenta las pautas de la World Wide Web Consortium (W3C).

- Estructura: La estructura de la página estará dividida en tres partes bien diferenciadas. En primer lugar, estará el encabezado en el que se encontrará la barra de navegación sobre una imagen de fondo que permita diferenciar esta de otras partes, siendo esta imagen diferente para distintas secciones de la página lo que permite identificar mejor la navegación. En el encabezado el tamaño de letra será mayor que en otros casos incluso con la fuente en negrita para facilitar la visualización y resaltar la importancia de ese contenido. Por otra parte, estará el cuerpo de la página en el que se dispondrá toda la información necesaria para que el usuario pueda interactuar con la página. Este apartado se encontrará sobre un fondo blanco para mejorar la visualización del contenido con variaciones de tipo y tamaño de letra en función del uso, como pueden ser títulos o contenido de artículos. Por último, estará el pie de página que permitirá poner fin al apartado de contenido, tomando un fondo de diferente color al anterior, en este caso negro.

- Colores: Otro factor importante para tener en cuenta para mejorar la visualización es el uso de los colores en la interfaz. Los colores principales que se utilizarán serán: blanco y negro, el blanco predominará en el fondo del cuerpo de la página de forma que haya mejor visualización de la información que se realizará en una fuente negra. El color negro estará presente en la mayoría de la información presentada al usuario y en el pie de página de las secciones para poner fin a esta. Por otra parte, las imágenes utilizadas en cada sección de la página, es decir, en las cabeceras y en el contenido de cada una de las secciones tratarán de seguir unas tonalidades similares dentro de la sección para dar una sensación de armonía y conjunto que resulte atractiva al usuario.
- Tipografía: En este caso se empleará una tipografía clásica que sea legible con facilidad con un tamaño adecuado para su lectura en los diferentes dispositivos en los que se pueda ejecutar la aplicación.
- Animaciones: Por otra parte, se emplearán animaciones utilizadas en el contenido del cuerpo de la página para incorporar dinamismo a la navegación por las diferentes secciones de la aplicación.

PROCESAMIENTO DE LOS DATOS

Por otra parte, es necesario desarrollar una herramienta que permita al usuario comprobar la validez de una posible ruta. Para ello será necesario emplear tecnologías de Infraestructuras de Datos Espaciales (IDEs) desplegadas por diferentes autoridades ya sean autonómicas o locales. Una IDE se trata de un conjunto de datos, metadatos, tecnologías, políticas, estándares, recursos humanos y usuarios, armonizados e integrados en un sistema virtual para compartir información geográfica en la red («IDE» 2020). En este caso las IDEs que serán necesarias para la aplicación son aquellas que pertenecen a las zonas en las que existen restricciones de vuelo, es decir, poblaciones, aeropuertos, zonas de especial protección de aves y parques naturales. Los recursos utilizados en este caso para el desarrollo de la aplicación serán el conjunto de datos pertenecientes a una IDE, estos datos están accesibles a través de las páginas oficiales de diferentes instituciones tanto estatales como autonómicas. Estos datos no cuentan con los servicios WPS y WFS necesarios

para realizar cálculos y procesamiento sobre los mismos por lo que será necesario obtener el conjunto de datos disponible para descarga. Uno de los inconvenientes que podría suponer esta forma de proceder sería utilizar datos desactualizados, aun así, en el caso de estos datos no sufren actualizaciones continuas debido a que no se producen grandes cambios, siendo la última actualización de estos en diciembre de 2020.

Estos datos se procesarán de distintas formas en función del tipo de restricción al vuelo de cada zona. En primer lugar, para realizar el procesamiento será necesario el cálculo de un encuadre centrado en la ruta dibujada para cada zona de estudio, de esta forma se evitará procesar datos innecesarios en las siguientes operaciones. Se realizará la comprobación de que existen lugares afectados de cada zona para proceder con el resto de las operaciones o en caso contrario terminar el procesamiento de esa zona de restricciones. Analizando cada zona por separado:

1. Aeropuertos: Tras comprobar si existen zonas de aeropuerto en el encuadre de la ruta se procederá a calcular la intersección con la ruta dibujada por el usuario y en este caso se almacenará el resultado en la base de datos. En el caso de que no existan zonas de aeropuertos en el encuadre se dejará de procesar la zona.
2. Poblaciones: En este caso se procederá de la misma manera que en el caso de los aeropuertos.
3. ZEPAs y Parques naturales: En primer lugar, se comprobará la existencia de zonas de ZEPAs y/o Parques Naturales en el encuadre de la ruta, esto dará lugar a cuatro casos diferentes:
 - a. Existen solo ZEPAs: en este caso se procederá al cálculo de la intersección de la ruta con la zona de la ZEPA afectada.
 - b. Existen solo Parques Naturales: se procederá al cálculo de la intersección con la zona de Parques Naturales afectada.
 - c. No existe ninguna de las dos zonas: en este caso se dejarán de procesar las dos zonas mencionadas.
 - d. Existen las dos zonas: En este caso se procederá a realizar la unión de las zonas de ZEPAs y Parques Naturales para procesarlos de forma única ya que forman parte de lugares con restricciones similares. Tras

realizar la unión se procederá a realizar la intersección con la ruta y posteriormente a almacenar los resultados en la base de datos.

En la figura siguiente se puede ver un esquema de la forma de procesamiento para obtener las zonas conflictivas en la ruta planificada por el usuario.

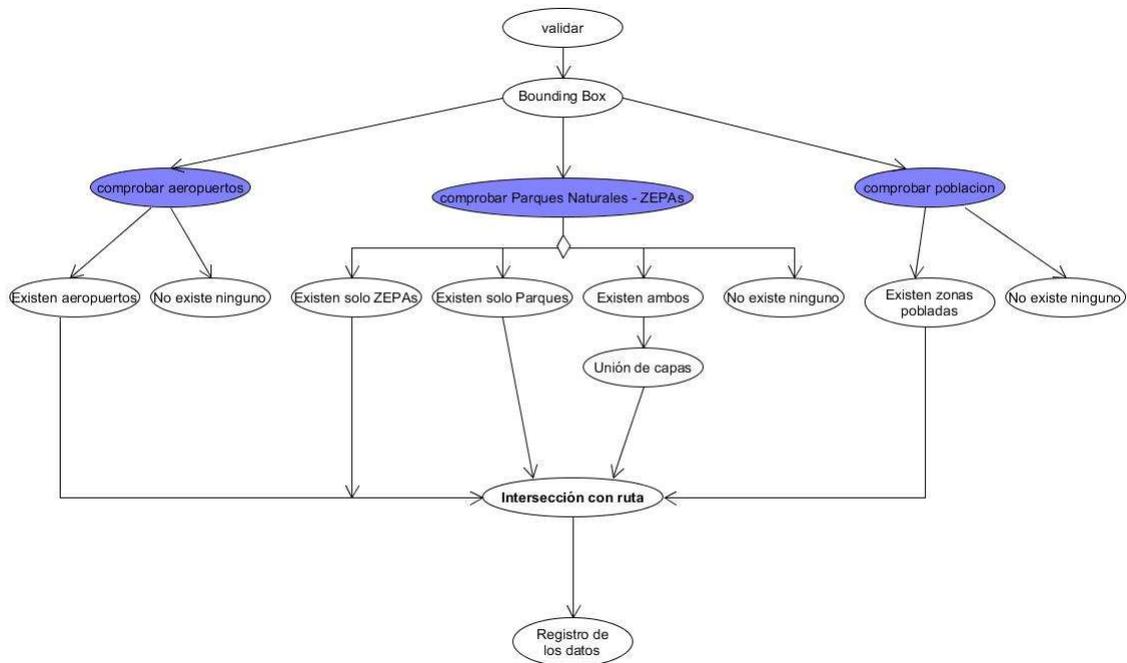


Figura 5: Esquema de procesamiento para la validación de la ruta

ACCIONES DEL USUARIO

El usuario tendrá la posibilidad de realizar diferentes acciones con el uso de la aplicación como son el registro o inicio de sesión para almacenar sus rutas, el dibujo o modificación de las rutas planificadas y la visualización de las diferentes zonas (zonas con limitaciones o rutas procedentes de usos anteriores de la aplicación).

1. Inicio de sesión: Para el inicio de sesión se empleará Firebase, Firebase se trata de un conjunto de herramientas de Google orientadas al desarrollo de aplicaciones en diferentes plataformas como son web o Android. En este caso se empleará la herramienta de autenticación para facilitar le procesamiento de datos personales de forma externa a la aplicación. De esta forma se podrá iniciar sesión de forma rápida y segura con gran cantidad de cuentas.
2. Dibujo: Para el dibujo de la ruta a planificar el usuario tendrá la opción de realizarla sobre el mapa de referencia. En este proceso se creará una colección de características en las que se añadirá la geometría donde se almacenará el

recorrido dibujado por el usuario (ver Figura 7).

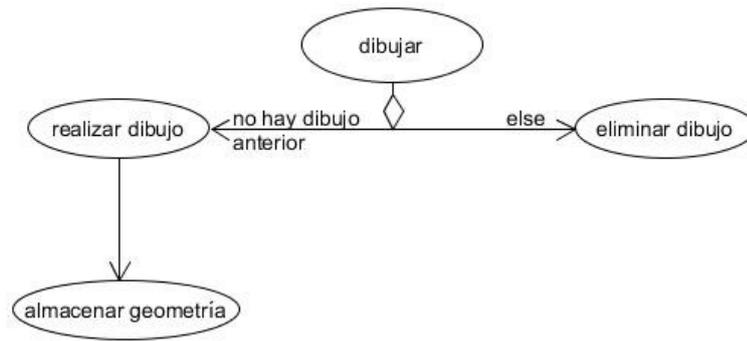


Figura 6: Esquema de procesamiento para la validación de la ruta.

3. Edición de la ruta: En el caso de que el usuario no este conforme con la ruta dibujada y antes de proceder al calculo de la validez de esta, el usuario podrá modificar el dibujo realizado. Para ello el usuario seleccionará una parte de la ruta y cambiarla de posición dando lugar a un cambio en la colección de características creadas anteriormente. (ver Figura 8).

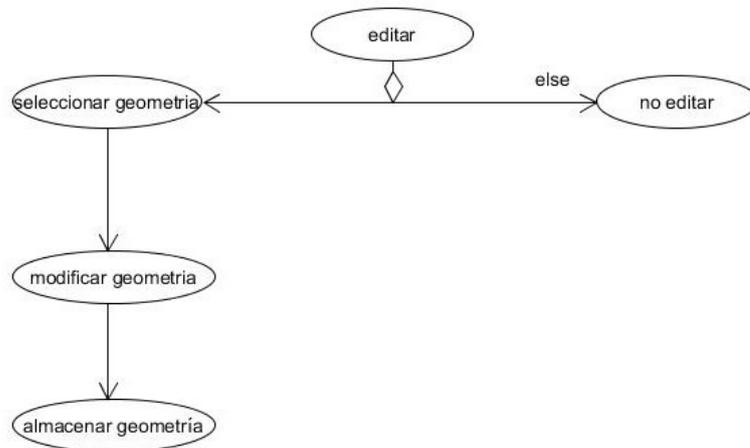


Figura 7: Esquema procedimiento para editar ruta.

Una vez vistas las principales acciones de forma particular podremos dar una visión general en el uso de la aplicación desde el punto de vista del usuario (ver Figura 9).

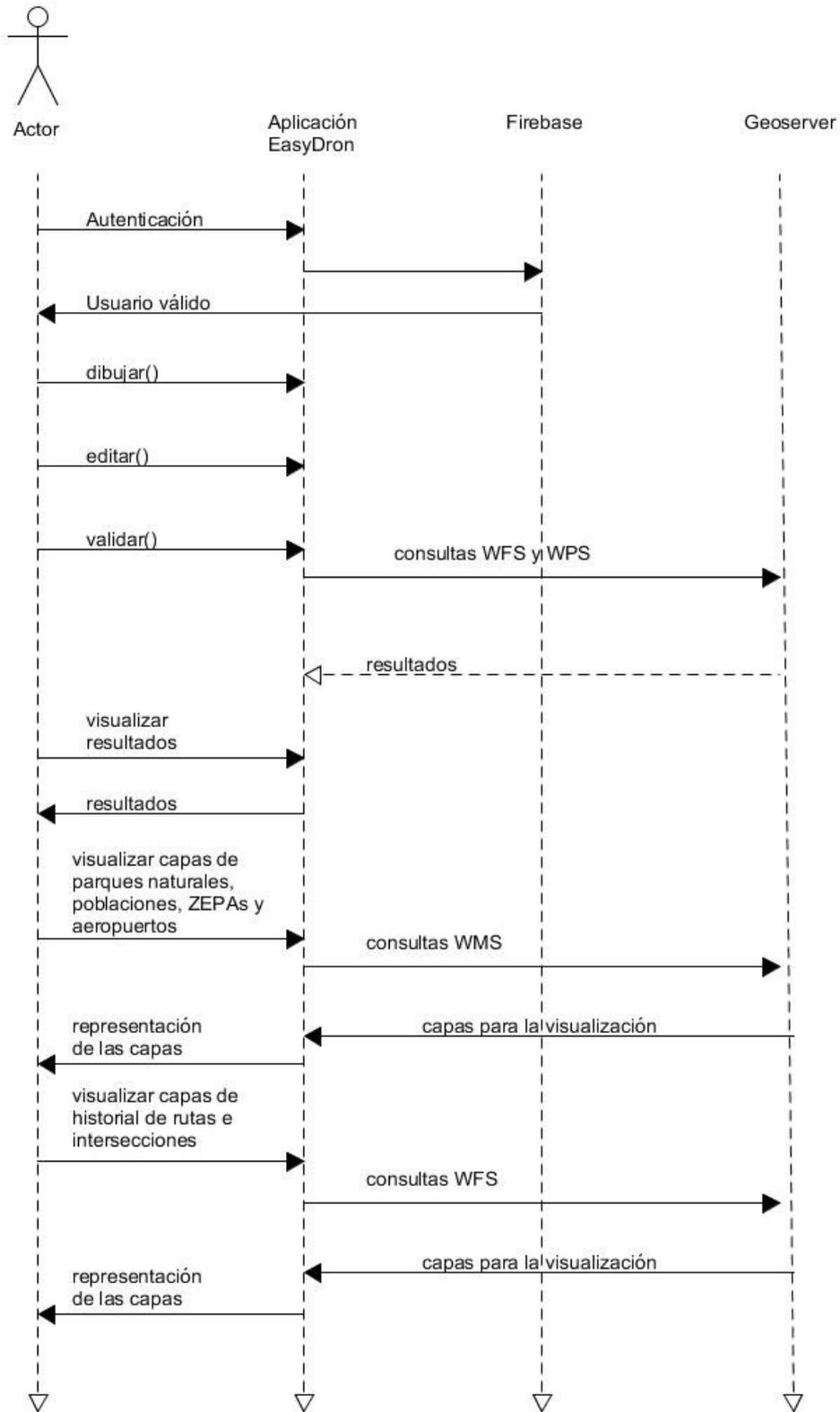


Figura 8: Esquema funcionamiento de la aplicación.

DESARROLLO DE LA APLICACIÓN

Para llevar a cabo el desarrollo de la aplicación es necesario el desarrollo tanto de una parte servidora y otra cliente. La parte del servidor será la encargada de almacenar y actualizar los datos necesarios y realizar las operaciones solicitadas por el cliente mediante consultas WPS, WFS o WMS para el cálculo de la validez de la ruta. En cuanto a la parte del cliente se encargará de mostrar al usuario una interfaz con la información necesaria y la aplicación para realizar la comprobación de las rutas junto con información relativa a las diferentes zonas.

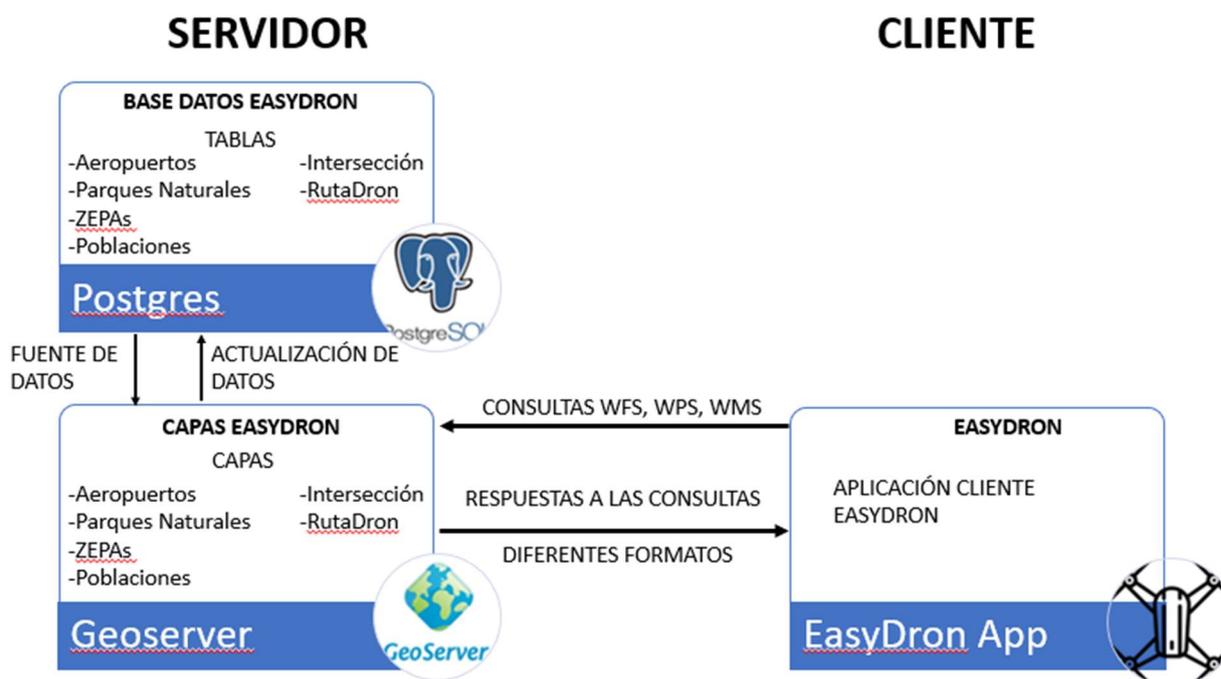


Figura 9: Esquema aplicación EasyDron.

SERVIDOR

En el servidor debemos tener en cuenta distintos factores como son los datos utilizados, la base de datos y el procesamiento de los datos.

DATOS

En primer lugar, tenemos que obtener los datos de las distintas zonas que queremos analizar, estos datos están disponibles en archivos en formatos .shp en los que se

almacenan tanto los datos espaciales como los datos textuales que cumplen la función informativa del conjunto de datos. En este caso los datos que necesitaremos son:

- Aeropuertos: Este recurso lo descargamos desde el Centro de Descargas del Centro Nacional de información Geográfica, accesible en el siguiente enlace: <http://centrodedescargas.cnig.es/CentroDescargas/buscadorCatalogo.do?codFamilia=REDTR>
Los datos que nos interesan de este recurso son: la geometría, la tipo de área (tip_area) que mediante un código numérico indica el tipo de emplazamiento aéreo, en este caso utilizaremos los aeródromos identificados con el 1. También resulta de interés el nombre de la pista (nombre) que aporta una descripción del aeródromo y la localización. Otro atributo a tener en cuenta es el responsable del aeródromo (fuented) que indica la autoridad responsable de las inmediaciones.
- Zonas de especial protección de aves: En este caso podemos encontrar estos datos en la página del Ministerio para la transición ecológica y el reto demográfico, accesible en: https://www.miteco.gob.es/es/biodiversidad/servicios/banco-datos-naturaleza/informacion-disponible/rednatura_2000_zepa_descargas.aspx
De estos datos en son de especial utilidad la geometría, el nombre de la zona junto con una pequeña descripción (site_name) y la comunidad autónoma responsable de la zona (ac).
- Parques naturales: Estos datos están disponibles para su descarga en la página web del Ministerio de agricultura, pesca y alimentación, accesible en: <https://www.mapama.gob.es/ide/metadatos/srv/spa/metadata.show?uuid=cb550555-548b-450f-bee1-4435a4c40b6c>
Los datos utilizados en este caso son la geometría, el nombre del parque natural (sitename) y la comunidad autónoma responsable (ccaa_n_enp).
- Zonas urbanas: Los datos de las poblaciones se encuentran disponibles en el Centro de Descargas del Centro Nacional de información Geográfica, accesible en el siguiente enlace: <http://centrodedescargas.cnig.es/CentroDescargas/catalogo.do?Serie=CAANE>
Los datos para tener en cuenta de las poblaciones son la geometría y el nombre de la población (nombre).

Todos estos datos que podemos descargar desde los diferentes enlaces se encuentran en formato shapefile (.shp). Este tipo de formatos es utilizado para almacenar la información geográfica de una ubicación junto con la información descriptiva necesaria para su comprensión. Las geometrías que se pueden representar mediante este formato son: líneas, puntos o polígonos (Morales 2013).

Una vez descargados estos archivos procedemos a realizar una simplificación de las geometrías almacenadas en estos debido a la alta carga de procesamiento del conjunto de datos original. Para ello será necesario el uso de un cliente GIS que nos permita realizar visualización y tratamiento de los datos incluidos en el archivo, en este caso el elegido es QGIS ya que es de código abierto y gratuito.

La simplificación que se realiza a los datos permite pasar de archivos de decenas o incluso cientos de megabytes a archivos de pocos megabytes o incluso kilobytes.

La ventaja de realizar este procesamiento de los datos es la reducción del tamaño de estos y por tanto un aumento en la velocidad de procesamiento de estos. Por el contrario, existe una desventaja que es la pérdida de precisión de los datos ya que las geometrías perderán exactitud respecto al original. Por esto es importante la selección del factor de simplificado adecuado para cada caso, en este caso se ha elegido un factor buscando el compromiso entre pérdida de precisión y reducción del tamaño de los datos.

Otro factor importante que hay que tener en cuenta es el sistema de referencia en que se encuentran estos datos, en este caso los datos de origen están en el sistema de referencia EPSG 4258 para las capas de aeropuertos, parques naturales y poblaciones y EPSG 25830 para las zonas de especial protección de aves (ZEPA). EPSG o European Petroleum Survey Group, organismo formado por expertos cartográficos y en geodesia que desarrollaron un repositorio de parámetros geodésicos y proyecciones cartográficas y geodesicas. El sistema EPSG 4258 se trata de una representación geodésica tridimensional centrada en Europa y cuya unidad de medida son los grados (Klokian Technologies 2018) lo que dificulta la realización de cálculos en términos de distancias. EPSG 25830 se trata igualmente de un sistema de referencia centrado en Europa en el huso 30 cuya unidad de medida es el metro, debido a esta disposición que

se puede observar en la Figura 10, en las zonas más exteriores de la península o islas como Canarias se producen más imprecisiones. (Klokkan Technologies 2000).



Figura 10: Sistema de referencia EPSG 25830

Debido a que se utilizan varias capas para el procesamiento necesitamos que estas capas tengan un sistema de referencia común. En nuestro caso interesa utilizar el sistema EPSG 3857 ya que se trata de un sistema de coordenadas proyectadas utilizado en aplicaciones de mapeo y representación basada en la web como puede ser Google Maps y cuya unidad de medida es el metro (Klokkan Technologies 2015) En este caso se trata de un sistema de coordenadas proyectadas, lo que significa que es un sistema plano por lo que simplifica la realización de cálculos de distancias, por la contra produce distorsiones debido a que se trata de una proyección de una superficie esférica o elíptica a una superficie plana. Se empleará este sistema ya que el objetivo de la aplicación es trazar rutas y ubicaciones para los cual resulta útil al ser un sistema proyectado al mismo tiempo que facilita los cálculos de distancias. Para ello utilizamos igualmente el cliente QGIS que nos permite realizar este proceso con facilidad. En primer lugar, tenemos que cargar las capas y utilizar la herramienta de gestión de datos llamada Reproyectar Capa de esta forma podemos convertir la capa al sistema de referencia deseado (ver Figura 11).

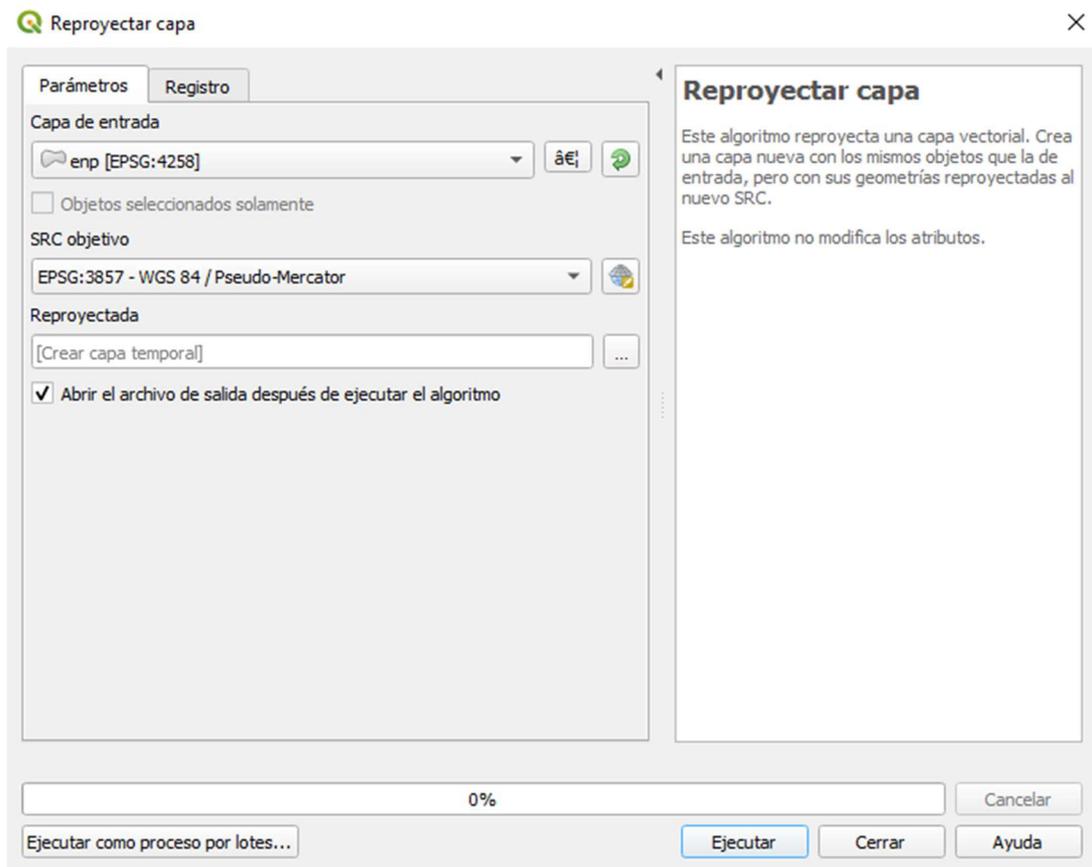


Figura 11: Reproyección capa Parques Naturales

Realizando este proceso con todas las capas necesarias tenemos un sistema de referencia común con unidades en metros. A continuación, se realiza la simplificación de las capas para que el procesamiento y tratamiento de los datos sea más eficaz.

QGIS nos ofrece una herramienta llamada Simplificar que nos permite realizar una simplificación de las geometrías de la capa. La simplificación realizada es de 10 metros como se puede ver en la Figura 12, se toma una medida de 10 metros ya que a pesar de que en el peor de los casos se 10 metros de precisión una medida bastante grande, el caso normal es la simplificación con medidas menores debido a las formas que toman las geometrías. Esto supone una mejora significativa en el procesamiento de los datos por parte del servidor a la hora de realizar los cálculos sin tener mucha pérdida de información respecto a la original.

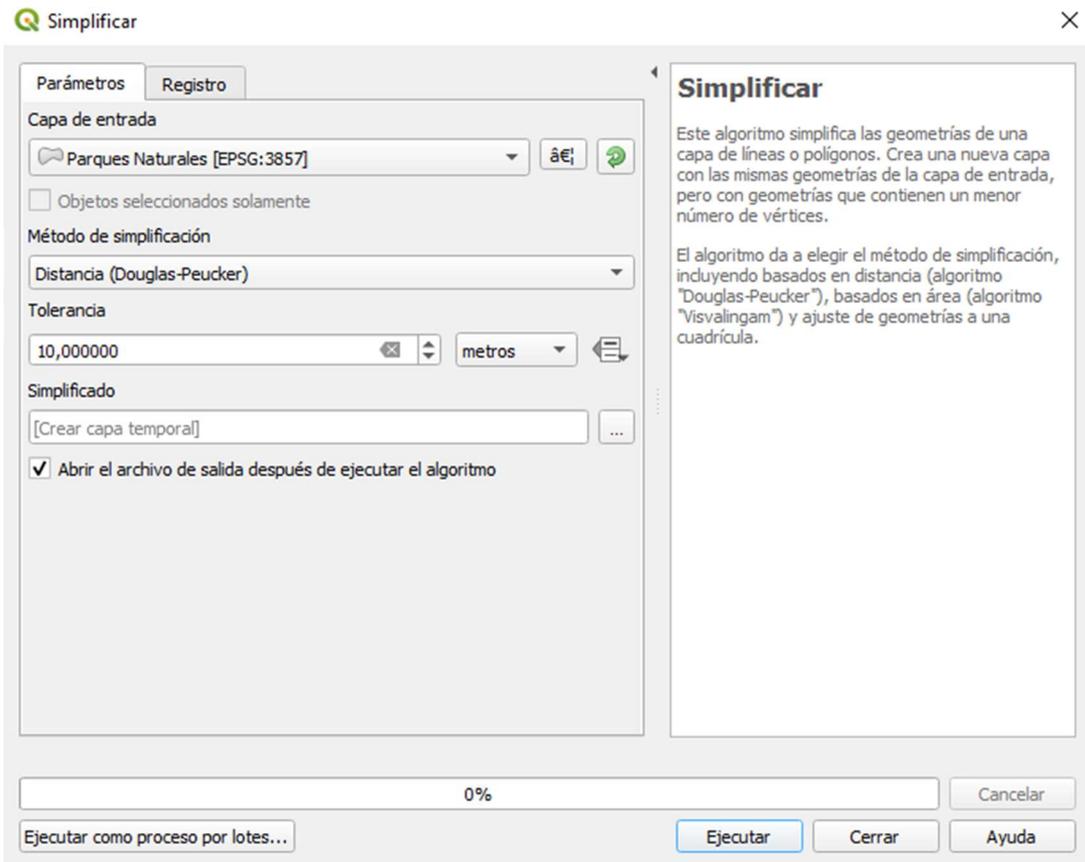


Figura 12: Simplificación capa Parques Naturales

Con esta simplificación conseguimos que archivo shapefile de la capa de parques naturales pase de ocupar 79.889 Kb a unos 11.000 Kb aproximadamente con una simplificación sin perder mucha información geográfica.

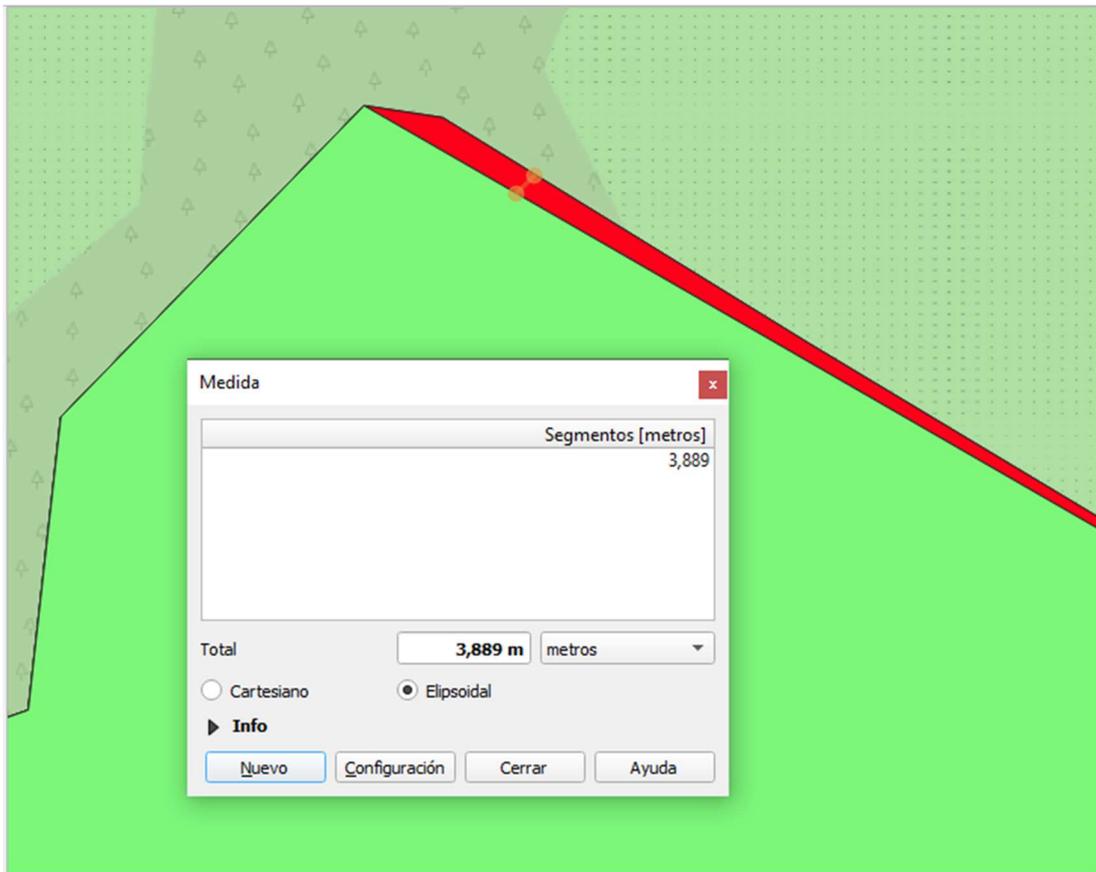


Figura 13: Distancia simplificada geometría de parques naturales.

Como podemos ver en la Figura 13 en una geometría que representa un espacio de parque natural (rojo sólido) y la simplificación de la capa de parques naturales (verde sólido) es prácticamente idéntica en gran parte, excepto alguna en el que la distancia es de unos 4 metros de diferencia. Una vez realizado este procesamiento de los datos ya podemos importarlos a la base de datos.

BASE DE DATOS

Para poder almacenar y utilizar estos datos junto con la aplicación del cliente necesitaremos un servidor de mapas y una base de datos espacial, en este apartado nos centraremos en la base de datos.

Para poder almacenar las áreas o geometrías tanto las disponibles en los archivos generados como las generadas mediante la aplicación del cliente es necesario disponer de un base de datos espacial. En este caso se utiliza PostgreSQL, que es un sistema gestor de datos relacionales de código abierto al que se puede añadir extensiones. (González Gil 2018), la extensión necesaria para poder procesar y almacenar datos

espaciales es PostGIS impulsada por OsGEO. Para facilitar el manejo y administración de los datos se utiliza el gestor PgAdmin4 que presenta una interfaz gráfica sencilla en la que interactuar con la base de datos.

En primer lugar, para comenzar a importar nuestros datos y crear las tablas necesarias, hay que crear una base de datos para ello basta con posicionarnos en “Databases” y con el botón derecho seleccionar “Create” y “Database”. A continuación, aparece un cuadro de diálogo en el que dar nombre a la base de datos, en este caso EasyDron, haciendo referencia al nombre de la aplicación. Con la base de datos ya creada es necesario atribuirle la característica de espacialidad, en el apartado de “Extensions” creamos una nueva extensión “Postgis”, de esta forma la base de datos está preparada para añadir los datos espaciales (ver Figura 14).

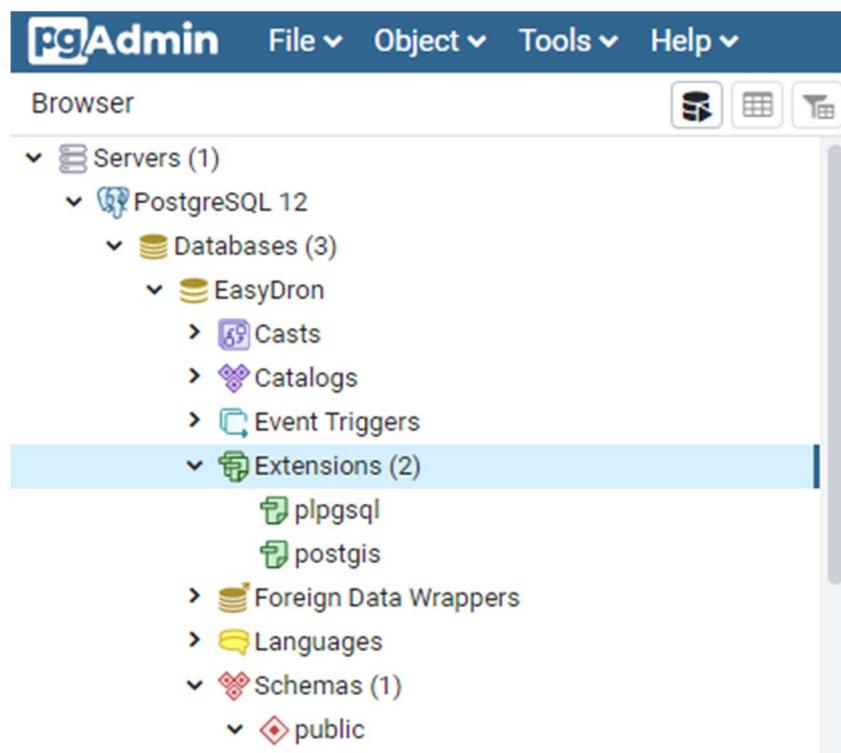


Figura 14: Base de datos espacial EasyDron

Para añadir los datos de los archivos en formato shapefile a la base de datos utilizamos el programa *shp2pgsql* que permite importar los datos con gran facilidad, para ello basta con incluir los datos para realizar la conexión con la base de datos, usuario creado

en pgAdmin4, contraseña y nombre de la base de datos (EasyDron), junto con los datos del servidor, el caso por defecto es el que se muestra en la Figura 15.

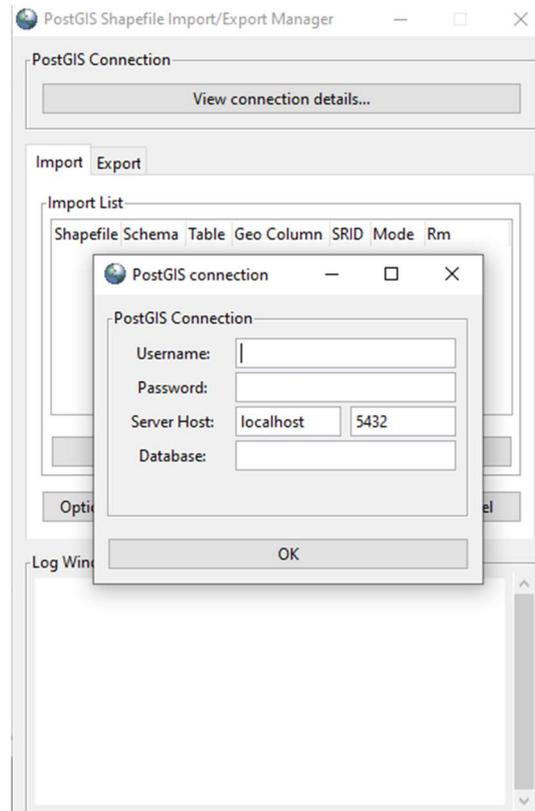


Figura 15: Datos conexión base datos

En “Add File” podemos seleccionar los archivos que se desean importar, en este caso será la capa de aeropuertos, zonas de especial protección de aves, parques naturales y poblaciones. En la columna “Table” escribimos el nombre que deseamos para la tabla de cada archivo, para el desarrollo de la aplicación los nombres elegidos son los siguientes: Aeropuertos, ZEPA, ParquesNaturales y Población. Otro campo que es necesario rellenar es el SRID, en el que se indica el sistema de referencia elegido, 3857 (ver Figura 16).

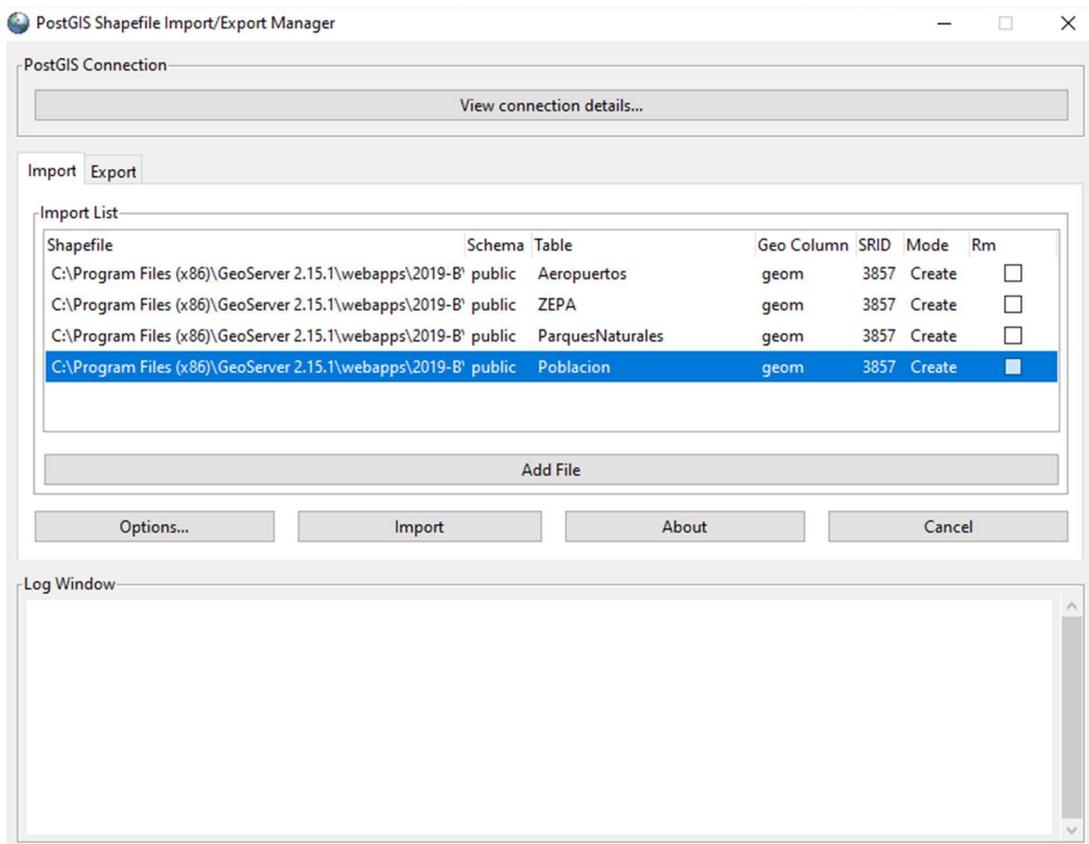
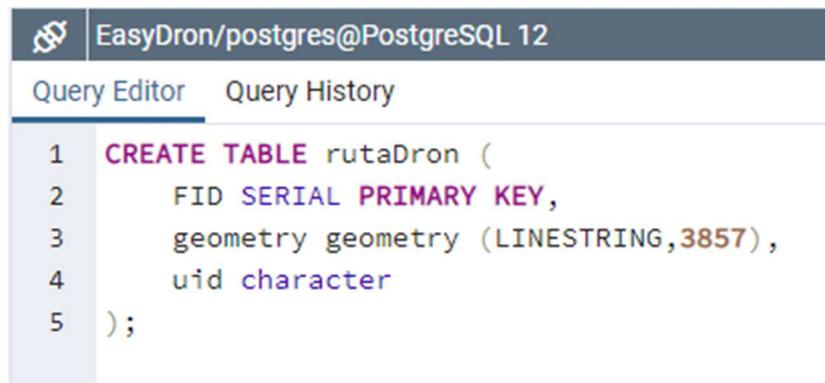


Figura 16: Importación de archivos a la base de datos.

Por otra parte, es necesario tratar con datos propios para poder realizar el cálculo de la validez de la ruta mediante la aplicación. Las tablas que se crean para este proceso son la tabla de la ruta a calcular y la tabla de la intersección.

CREACIÓN TABLA RUTA PLANIFICADA DEL DRON

Para la creación de las tablas se realiza mediante una consulta SQL de inserción como la que se puede ver en la Figura 17, empleando la herramienta de consultas SQL que proporciona pgAdmin4.



```
EasyDron/postgres@PostgreSQL 12
Query Editor Query History
1 CREATE TABLE rutaDron (
2     FID SERIAL PRIMARY KEY,
3     geometry geometry (LINESTRING,3857),
4     uid character
5 );
```

Figura 17: Creación tabla rutaDron

Los campos necesarios de la tabla son el FID como identificador único de la ruta, y GEOMETRY de tipo línea (geometría) en el sistema de referencia utilizado en toda la aplicación (3857) y por último un identificador del usuario que ha realizado la ruta.

CREACIÓN TABLA INTERSECCIÓN DE LA RUTA CON LAS ZONAS RESTRINGIDAS

También es necesario la creación una tabla llamada intersección para almacenar el resultado del cálculo de la intersección de la capa de la ruta del dron con las capas de aeropuertos, parques naturales, zonas de especial protección de aves y las poblaciones. Los campos de esta tabla son los mismos campos que se encuentran en las tablas de los datos descargados.



```
1 CREATE TABLE interseccion(  
2     id SERIAL PRIMARY KEY,  
3     idRuta double precision,,  
4     --Aeropuertos  
5     Aeropuertos_id_aerodro double precision,  
6     Aeropuertos_id_area double precision,  
7     Aeropuertos_tip_area int,  
8     Aeropuertos_tip_aread char(60),  
9     Aeropuertos_desigandor char(20),  
10    Aeropuertos_nombre char(200),  
11    Aeropuertos_longitud numeric,  
12    Aeropuertos_longitudd char(20),  
13    Aeropuertos_anchura numeric,  
14    Aeropuertos_anchurad char(20),  
15    Aeropuertos_comp_sup int,  
16    Aeropuertos_comp_supd char(20),  
17    Aeropuertos_tip_pista numeric,  
18    Aeropuertos_tip_pistad char(20),  
19    Aeropuertos_fuente int,  
20    Aeropuertos_fuented char(40),  
21    Aeropuertos_fecha_alta date,  
22    --Zona de Especial Proteccion de Aves  
23    ZEPA_site_code char(20),  
24    ZEPA_site_name char(100),  
25    ZEPA_ac char(150),  
26    ZEPA_hectareas double precision,  
27    ZEPA_shape_leng double precision,  
28    ZEPA_shape_area double precision,  
29    --Parques Naturales  
30    ZEPA_sitecode char(20),  
31    ZEPA_sitename char(254),  
32    ZEPA_figura_lp char(100),  
33    ZEPA_area_ha double precision,  
34    ZEPA_year_a char(4),  
35    ZEPA_ccaa double precision,  
36    ZEPA_enp_fgral double precision,
```

```

37     ZEPA_ccaa_n_enp char(254),
38     ZEPA_shape_len double precision,
39     ZEPA_layer char(100),
40     ZEPA_path_a char(254),
41     --Poblaciones
42     Poblacion_id_ep bigint,
43     Poblacion_nombre char(200),
44     Poblacion_codine char(11),
45     Poblacion_ine int,
46     Poblacion_fecharef int,
47     Poblacion_habitantes bigint,
48     Poblacion_capital char(4),
49
50
51     geom geometry(MULTILINESTRING,3857)
52
53 );

```

Figura 18: Creación tabla intersección.

Como se puede observar en los campos de cada capa con la que se realiza la intersección se incluye el nombre de la tabla. En el caso de los Aeropuertos todos los campos relacionados comienzan por “Aeropuertos_”, en los campos de zona de especial protección de aves y parques naturales los campos comienzan por “ZEPA_” y los campos de población por “Poblacion_”. Este cambio del nombre de los campos respecto a la capa original se debe a los diferentes procesos que sufren las capas durante el cálculo de la ruta, posteriormente se analizará cada uno de estos procesos.

SERVIDOR DE MAPAS

Otra de las herramientas necesarias para el desarrollo de la aplicación es un servidor de mapas que proporcione los mapas y datos en diferentes formatos como son WMS, WFS, WPS y WCS y permita realizar las consultas necesarias para el procesamiento de la aplicación. El servidor elegido es Geoserver, el motivo de la elección es la facilidad de uso y al igual que el resto de las herramientas usadas es gratuito. Este servidor incluye los servicios WMS, WFS y WCS inicialmente con la instalación básica, el servicio WPS puede ser instalado con facilidad obteniendo el servicio de la web oficial de Geoserver.

- Web Feature Service (WFS): Estándar de interface OGC que define las operaciones web para la consulta y edición de las entidades geográficas. Este estándar permite al cliente consultar la colección de entidades geográficas (*GetCapabilities*), describir los atributos disponibles en cada entidad (*DescribeFeatureType*), consultar una colección de entidades en función de un filtro (*GetFeature*) y añadir, borrar o editar entidades (*Transaction*) («Web Feature Service | OGC» s. f.).
- Web Map Service (WMS): Este servicio permite realizar consultas mediante HTTP para solicitar imágenes de mapas desde bases de datos espaciales. En la consulta se incluye la capa y zona de interés y la respuesta estará formada por una o varias imágenes («Web Map Service | OGC» s. f.).
- Web Proccesing Service (WPS): Servicio que permite la realización de transformaciones y procesamiento de una o varias capas geoespaciales. Define como un cliente puede solicitar la ejecución de un proceso y como se maneja la salida («Web Processing Service | OGC» s. f.).

Para poder disponer de los datos incluidos en la base de datos mencionada anteriormente es necesario en primer lugar crear un espacio de trabajo, para ello hay que dar un nombre a este y un URI que identificara el acceso a los recursos.

Editar espacio de trabajo

Editar un espacio de trabajo existente

Nombre

URI del espacio de nombres

El URI del espacio de nombres asociado con este espacio de trabajo

Espacio de trabajo por defecto

Isolated Workspace

Configuración

Habilitado

Servicios

- WMTS
- WCS
- WFS
- WMS
- WPS

Figura 19: Creación de espacio de trabajo en Geoserver

A continuación, se crea un almacén de trabajo, que tiene que ser del tipo *Postgis* para poder enlazar con la base de datos, es necesario rellenar los datos relacionados con el nuevo almacén de trabajo y los datos necesarios para la conexión con la base de datos (ver Figura20).

Editar un origen de datos vectorial existente

PostGIS
PostGIS Database

Información básica del almacén

Espacio de trabajo *

EasyDron ▼

Nombre del origen de datos *

EasyDron

Descripción

Habilitado

Parámetros de conexión

host *

localhost

port *

5432

database

EasyDron

schema

public

user *

postgres

passwd

Espacio de nombres *

http://EasyDron.es

Figura 20: Almacén de trabajo en Geoserver

Una vez esta creado el almacén de trabajo solamente queda publicar las capas que tenemos almacenadas en la base de datos para que puedan ser accesibles desde la aplicación. En la publicación es necesario rellenar los campos de “Encuadre nativo” para ello elegimos la opción “*Compute from SRS bounds*” y el campo “Encuadre Lat/Lon” seleccionando la opción “Calcular desde el encuadre nativo” (ver Figura 21).

EasyDron:Aeropuertos

Configure el recurso y la información de publicación para esta capa

Datos **Publicación** Dimensiones Cacheado de Teselas

Editar capa

Información básica del recurso

Nombre

Habilitado
 Anunciado

Título

Resumen

Palabras clave

Palabras clave actuales

Nueva palabra clave

Vocabulario

Sistema de referencia de coordenadas

SRS nativo

SRS declarado

Gestión de SRC

Encuadres

Encuadre nativo

Min X	Min Y	Máx X	Máx Y
-20.037.508,34278	-20.048.966,10401	20.037.508,34278	20.048.966,10401

[Calcular desde los datos](#)
[Compute from SRS bounds](#)

Encuadre Lat/Lon

Min X	Min Y	Máx X	Máx Y
-180,000000000000	-85,0599999999999	180,000000000000	85,0599999999999

[Calcular desde el encuadre nativo](#)

Control de geometrías curvas

Geometrías lineales pueden contener arcos circulares

Figura 21: Publicación capa de aeropuertos.

Una vez publicada todas las capas ya están disponibles tanto para su visualización como para la realización de distintos modificaciones o procesado de los datos y también se puede acceder a estas capas desde la aplicación del cliente.

Para mejorar la presentación en la aplicación de las capas almacenadas Geoserver ofrece la posibilidad de crear estilos para las diferentes capas, en este caso se tratan de capas con formas poligonales en las que es interesante cambiar únicamente el color de

la representación. Para ello en la opción “Estilos” Geoserver permite copiar el código XML de estilos predeterminados y modificarlos para crear nuevos, en este caso utilizamos el estilo de polígonos para cambiar el color y de esta forma diferenciar las capas utilizadas mediante distintos colores.

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <StyledLayerDescriptor version="1.0.0"
3   xsi:schemaLocation="http://www.opengis.net/sld StyledLayerDescriptor.xsd"
4   xmlns="http://www.opengis.net/sld"
5   xmlns:ogc="http://www.opengis.net/ogc"
6   xmlns:xlink="http://www.w3.org/1999/xlink"
7   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
8   <!-- a Named Layer is the basic building block of an SLD document -->
9   <NamedLayer>
10    <Name>Aeropuertos</Name>
11    <UserStyle>
12      <!-- Styles can have names, titles and abstracts -->
13      <Title>Default Polygon</Title>
14      <Abstract>A sample style that draws a polygon</Abstract>
15      <!-- FeatureTypeStyles describe how to render different features -->
16      <!-- A FeatureTypeStyle for rendering polygons -->
17      <FeatureTypeStyle>
18        <Rule>
19          <Name>rule1</Name>
20          <Title>Gray Polygon with Black Outline</Title>
21          <Abstract>A polygon with a gray fill and a 1 pixel black outline</Abstract>
22          <PolygonSymbolizer>
23            <Fill>
24              <CssParameter name="fill">#F2A82E</CssParameter>
25            </Fill>
26            <Stroke>
27              <CssParameter name="stroke">#F2A82E</CssParameter>
28              <CssParameter name="stroke-width">1</CssParameter>
29            </Stroke>
30          </PolygonSymbolizer>
31        </Rule>
32      </FeatureTypeStyle>
33    </UserStyle>

```

Validar Apply Enviar Cancelar

Figura 22: Modificación de estilos con Geoserver.

Como podemos observar en el XML hay varios parámetros CSS que podemos modificar, los parámetros necesarios para modificar el color son:

`<CssParameter name="fill">` y `<CssParameter name="stroke">` que hacen referencia al relleno y al borde del polígono respectivamente. El código hexadecimal que le precede hace referencia al código de color utilizado, para esta aplicación los colores utilizados son los siguientes:

- Parques Naturales: Verde - #93CF93
- ZEPAs: Azul - #58E3E3
- Aeropuertos: Naranja - #F2A82E
- Poblaciones: Gris - #596259

De esta forma tendremos un código de colores para capa que nos permitirá diferenciar fácilmente que capa estamos visualizando y que polígono pertenece a cada capa.

CLIENTE

En el apartado del cliente se incluye la página web que suministra la estructura y contenido relacionado con la aplicación y la aplicación. La página web permite al usuario tener un contexto del momento actual de los drones y una aproximación a la normativa vigente incluida en el Real Decreto 1036/2017 permitiendo así el conocimiento de las limitaciones que impone al vuelo con dron. Por otra parte, se encuentra la aplicación con la que el usuario puede interactuar para obtener información de la ruta que desea realizar.

SOPORTE WEB

En este apartado se encuentran un conjunto de archivos que permiten dar estructura, estilo soporte e información a la aplicación para que resulte atractivo al usuario.

Una de las características más importantes, que a su vez es una pauta para el desarrollo de aplicaciones web dada por el World Wide Web Consortium (W3C) apartado es la capacidad de responsividad, es decir la capacidad de adaptar su contenido a diferentes dispositivos de tal forma que se muestre de forma adecuada en cada caso («Web Content Accessibility Guidelines (WCAG) 2.1» s. f.).

Para conseguir esta característica se ha empleado el *framework* Bootstrap en su última versión. Bootstrap fue desarrollado por Twitter en 2010, que posteriormente se extendería como una herramienta para el desarrollo de páginas y aplicaciones web facilitando el diseño y ofreciendo nuevas oportunidades de creación (Mark Otto, Jacob Thornton, and Bootstrap s. f.).

Para poder utilizar este *framework* en el desarrollo de la página web en primer lugar es necesario descargar tanto los archivos del propio Bootstrap como el archivo JQuery, ya que muchos componentes de Bootstrap requieren el uso de JavaScript. Estos archivos se pueden encontrar en las páginas oficiales de ambos de forma gratuita.

Otra librería que se ha empleado junto con Bootstrap es Animate.css que permite crear animaciones en diferentes partes del documento lo que genera mayor sensación de dinamismo. Esta librería es accesible a través de la página oficial en la que se indican los pasos de cómo utilizarla de forma gratuita.

Para que estas animaciones puedan utilizarse en ciertos casos como en el caso que se quieren activar mientras se desplaza verticalmente a través de la página es necesario

el uso de WOW.JS. WOW.JS es un plugin de JavaScript utilizado en conjunto con Animate.css para la configuración de la animación: forma en la que se despliega, estilos, duración, velocidad con la que se ejecuta... Para la instalación de este plugin es necesario descargar el paquete disponible en su página web y vincular el estilo en la hoja HTML. Por otra parte, es necesario incluir el script al inicio del documento para iniciar el plugin.

```
<script>  
    new WOW().init();  
</script>
```

EasyDron posee varios archivos en los que se presenta distinta información al usuario, todos ellos están desarrollados en el lenguaje de marcados HTML, junto con CSS para aportar estilo y JavaScript para aportar distintas funcionalidades. La estética elegida trata de ser simple y minimalista ya que es una forma de presentar la información de forma clara al usuario además de ser una tendencia actual en el desarrollo de páginas web. Los archivos desarrollados son los siguientes:

INDEX.HTML

Es archivo se trata de la página inicial, en la que se da a conocer la aplicación y da la posibilidad de conocer diversa información acerca de los drones y normativa.

En primer lugar, nos encontramos con una cabecera común a las distintas páginas en la que se encuentra la barra de navegación que permite acceder a diferentes secciones de la página web junto con una imagen que es diferente en función de la sección.



Figura 23: Cabecera EasyDron

Como se puede ver en la figura 23 la barra de navegación incluye la sección de la aplicación (accesible tanto en el nombre de esta como en el botón situado a la derecha) y las secciones de inicio, preguntas y respuestas frecuentes y sobre nosotros. En el caso de utilizar un dispositivo móvil también se podrá acceder a la navegación, en este caso mediante el botón situado en la esquina superior derecha como se puede observar en la Figura 24.

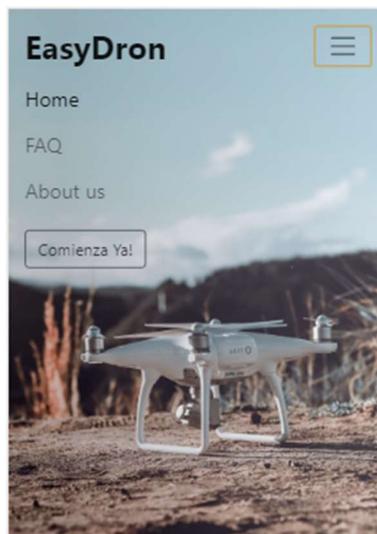


Figura 24: Cabecera EasyDron en dispositivo móvil

Esto se consigue mediante el uso de las clases y estilos incluidos en el *framework* de Bootstrap como puede verse en el siguiente fragmento de código.

```
<!--Cabecera con barra de navegación-->
<header class="CabeceraFoto" style="background-image:
url (fotos/index.jpg) ;">
  <nav class="navbar navbar-expand-md navbar-light bg-light ">
    <a class="navbar-brand"
href="app.html"><strong>EasyDron</strong></a>
    <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#collapsibleNavbar"
aria-controls="navbarSupportedContent" aria-
expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse"
id="collapsibleNavbar">
      <ul class="navbar-nav mr-auto">
        <li class="nav-item active">
          <a class="nav-link" href="Index.html">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="FAQ.html">FAQ</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="AboutUs.html">About us</a>
        </li>
      </ul>
    </div>
    <form class="form-inline my-2 my-lg-0">
```

```

        <a href="app.html" class="btn btn-outline-dark my-2 my-
sm-0" role="button" aria-pressed="true">Comienza Ya!</a>
    </form>
</div>
</nav>
</header>

```

Mediante el uso de la clase *navbar* junto con *navbar-expand-md* permite que la barra de navegación se expanda cuando se trata de una pantalla de tamaño medio y se reduzca pasando a ser un botón expandible en caso de que el tamaño sea menor como se puede observar en las Figuras 23 y 24 respectivamente. En esta página en primer lugar, se puede ver una pequeña presentación de la aplicación y la invitación al usuario a utilizarla y a continuación información relacionada con el ámbito de los drones. Esta información es una explicación y justificación de la importancia de EasyDron para realizar un vuelo seguro, también se presentan varios enlaces, uno de ellos al Real Decreto 1036/2017 para que el usuario pueda acceder fácilmente y obtener toda la información de la competencia reguladora. Por otra parte, se muestra noticias de interés del mundo de los drones para que el usuario pueda ser participe a un mayor nivel (ver Figura 25).

Comprueba tu nueva ruta

¿Estas pensando en realizar un vuelo con tu dron? Puedes comprobar si la zona elegida es apta, para ello solo tienes que iniciar la aplicación y seleccionar la ubicación o la ruta estimada para el vuelo.

¡Inicia ya la comprobación!

Comienza Ya!



Nuestro proyecto

Te invitamos a conocer nuestro proyecto y comprueba que aporta al sector de los drones.

Saber más



Noticia del día

Noticias interesantes relacionadas con el mundo de los Drones

Saber más



Conoce la normativa

Conoce la normativa de vuelos de Drones con el Real Decreto 1036/2017

Saber más

Figura 15: Contenido de presentación de la página Index

FAQ.HTML

En este apartado se presentan preguntas y respuestas que pueden surgir a cualquier usuario, se trata de una serie de dudas de forma genérica. En caso de que la duda del usuario se encuentre fuera de esa batería de dudas, se ofrece la posibilidad de un contacto al que enviar su pregunta. Para la visualización se emplea una lista en forma

de acordeón en la que el usuario puede seleccionar la pregunta de la que desea obtener una respuesta y esta se desplegará. Esta forma de visualización permite la información que se muestra no sea excesiva y pueda provocar confusión al usuario.

Preguntas y respuestas frecuentes

▼ ¿Como dibujar una ruta?
▼ ¿Como puedo ver las zonas con limitacion al vuelo?
Para visualizar las zonas en las que existe prohibición de vuelo puede seleccionar las distintas capas en el botón "Capas". Puede utilizar la visualización de estas zonas para planear su ruta.
▼ ¿Como se si mi ruta es segura?
▼ No consigo dibujar una ruta. ¿Que puedo hacer?
▼ No entiendo los resultados. ¿Que puedo hacer?
▼ ¿Cuantas rutas puedo consultar?
▼ ¿Tienes más dudas? Contáctanos

Figura 26: Contenido página preguntas y respuestas frecuentes.

ABOUTUS.HTML

Esta sección sirve para dar a conocer al usuario el contexto en el que se desarrolla la aplicación, el motivo por el que surge la creación de esta y cuáles son los beneficios que ofrece al usuario que la utiliza. La información se presenta en tres bloques: Actualidad, Aumento de accidentes y EasyDron. En la sección Actualidad se ofrece una pequeña descripción de la situación actual de la tecnología de los drones y el crecimiento que ha tenido lugar y se espera que siga teniendo. Aumento de accidentes presenta información relativa a los problemas que causa un uso inadecuado de los drones, dando a conocer algunas de las limitaciones que presenta la normativa a la hora de realizar un vuelo y como el desconocimiento puede causar incidentes en varios ámbitos. Por último, se muestra una descripción de como EasyDron puede solventar o contribuir en la mejora de estas situaciones invitando al posible usuario a utilizarla remarcando la facilidad de su uso (ver Figura27).

About Us

Actualidad

El número de drones se ha visto incrementado en los últimos años dando lugar a una nueva tecnología que despierta gran interés. Uno de los motivos del crecimiento del uso es la reducción del coste y la mejora tecnológica que han sufrido en los últimos años. Otro factor que ha propulsado el aumento es el uso en el sector laboral abriendo nuevas posibilidades de trabajo y una disminución de los riesgos en trabajos ya concidos. El otro sector que ha tenido gran acogida de estos dispositivos, es el recreativo. Cada vez más aficionados se hacen con un dron por distintos motivos: realización de fotografías de gran calidad y tomas imposibles sin su uso, grabación de vídeos o únicamente por el placer de pilotar uno. Los drones han demostrado ser una tecnología en crecimiento y que se espera que lo siga haciendo en los próximos años.



Aumento de incidentes

Debido al aumento del uso de los drones también se ha incrementado el número de incidentes causados por estos. La mayoría de estos incidentes están relacionados con la prohibición del vuelo en diferentes zonas, estas zonas son: Proximidades de los aeropuertos y espacio aéreo restringido, zonas de especial protección de aves (ZEPA), parques naturales o poblaciones (limitado por el peso del dron). Estas limitaciones están incluidas en el Real Decreto 1036/2017 que regula el vuelo de drones, el incumplimiento de esta puede dar lugar a distintos tipos de sanciones. En España esto ha propiciado la creación de una patrulla dedicada a controlar las infracciones de estas aeronaves y a realizar formaciones para un uso adecuado.

EasyDron

La aplicación **EasyDron** permite acercar la normativa y facilitar el cumplimiento de esta. De esta forma los usuarios podrán realizar vuelos con su dron con mayor seguridad evitando zonas conflictivas, sin necesidad de conocer la zona. Para ello solamente es necesario iniciar **EasyDron** y dibujar la próxima ruta de vuelo siguiendo las instrucciones, mediante el algoritmo de la aplicación se realizarán los cálculos pertinentes para indicar al usuario si se encuentran limitaciones al vuelo en la zona indicada o puede volar con normalidad. Tras realizar el cálculo se mostrará al usuario de forma clara una lista con las distintas zonas que pueden suponer una limitación y una conclusión sobre la posibilidad de realizar la ruta con su dron.



Figura 27: Contenido de la página AboutUs

APP.HTML

En este apartado nos encontramos con la aplicación con la que puede interactuar el usuario para comprobar la validez de una ruta, para ello se dispone de un mapa y una serie de opciones para que el usuario pueda llevar a cabo diferentes tareas. En primer lugar, antes de comenzar a estar disponible la aplicación se pide al usuario que se identifique o registre para poder utilizarla (ver Figuras 28).

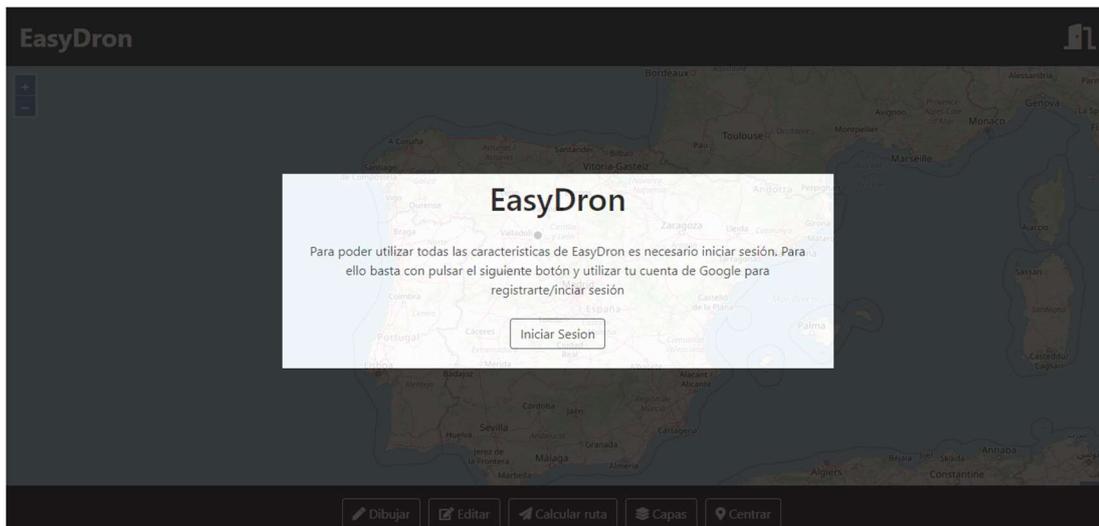


Figura 28: Petición inicio/registro.

A continuación, pulsando en el enlace se redirigirá a una página externa perteneciente a *Firebase* desde la que el usuario podrá iniciar sesión con su cuenta de Google. Para ello será necesario incluir la función *login()* llamada desde el botón que se observa en la figura 28, esta función contendrá la configuración de la herramienta de *Firebase* y la creación y obtención de datos relativos al usuario.

```
function login() {
    var provider = new firebase.auth.GoogleAuthProvider();

    firebase.auth().signInWithPopup(provider).then(function(user) {
        user = firebase.auth().currentUser;

        if (user != null) {
            uid = user.uid; // The user's ID, unique to the
            // Firebase project. Do NOT use
            // this value to authenticate with your backend server, if
            // you have one. Use User.getToken() instead.
        }

        historialinterseccion();
        historialrutas();
    })

    $(".bloqueador").fadeOut("slow");
}

```

En esta función *login()* también podemos observar la llamada a dos funciones: *historialintersección()* e *historialrutas()*, estas funciones son las encargadas de realizar una consulta WFS para obtener el historial de intersecciones y rutas del usuario y crear la capa para mostrarla al usuario.

```
async function historialrutas() {
```

```

        //consulta WFS para obtener las rutas dibujadas por
el usuario
        var historial_ruta=`<wfs:GetFeature service="WFS"
version="1.1.0" outputFormat="application/json"
xmlns:EasyDron="http://EasyDron.es"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs
http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">
    <wfs:Query typeName="EasyDron:rutaDron">
        <ogc:Filter>
            <ogc:PropertyIsEqualTo>
                <ogc:PropertyName>uid</ogc:PropertyName>
                <ogc:Literal>${uid}</ogc:Literal>
            </ogc:PropertyIsEqualTo>
        </ogc:Filter>
    </wfs:Query>
</wfs:GetFeature>`;

        // then post the request and add the received
features to a layer
        fetch("/geoserver/EasyDron/wfs", {
            method: "POST",
            headers: {
                "Content-Type": "application/xml;
charset=UTF-8"
            },
            body: historial_ruta
        }).then(function(response) {
            return response.json();
        }).then(function(json) {
            var features = new
ol.format.GeoJSON().readFeatures(json);
            HistorialRutaSource.clear();

            HistorialRutaSource.addFeatures(features);

        });
    }
}

```

Una vez iniciada la sesión el usuario tendrá acceso a la aplicación como se observa en la Figura 29.

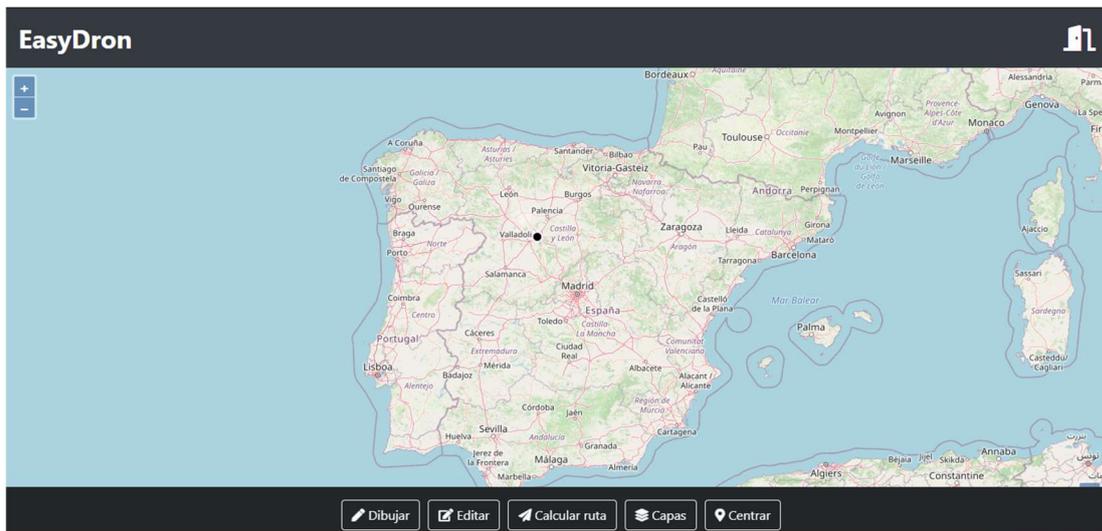


Figura 29: Interfaz de la aplicación EasyDron

En primer lugar, como se puede observar en la Figura 29 vemos una diferencia respecto a la barra de navegación. En este caso se elimina la imagen de fondo para mostrar el conjunto de la aplicación en el espacio de pantalla disponible sin tener que realizar una navegación a través de la página para facilitar el uso. También se incluye una nueva opción que es “Instrucciones” en la que se dan pautas al usuario para un buen uso de la aplicación, estas instrucciones incluyen: instrucciones de uso, comprensión de los resultados y recomendaciones para el vuelo (se entrará en detalle a lo largo del documento). Además, se dispone de un botón con un icono de una puerta que permite al usuario cerrar sesión y salir de la aplicación, para ello se llama a la función *logout()*.

```

async function logout () {

    await firebase.auth().signOut().then(function () {
        user = firebase.auth().currentUser;
        if(user != null){

        }
        else{
            uid = null;
            alert("Gracias por utilizar EasyDron, hasta
pronto!");
        }

        window.location.href = "./Index.html";
    }).catch(function(error) {
        alert("no se ha podido cerrar la sesión");
    });
}

```

A continuación se muestra un mapa de Open Layers, Open Layers es un proyecto que facilita el uso de mapas dinámicos en las webs de forma gratuita y con la posibilidad de mostrar una gran variedad de información geográfica en distintos formatos todo ello mediante el uso de JavaScript («OpenLayers — OSGeoLive 13.0 Documentation» s. f.) .

Para poder mostrar el mapa en la página web es necesario la creación de este, la asignación de estilos para una correcta visualización y la selección del tipo de mapa que se desea mostrar. En este caso hay dos mapas disponibles para la visualización, el que se observa en la Figura 29 y otro que muestra el mapa como una imagen satelital.

```
function initmap() {  
  
    /*-----Styles-----  
    -----*/  
    var text = new ol.style.Text({  
        textAlign: 'center',  
        scale: 1.3,  
        fill: new ol.style.Fill({  
            color: '#fff'  
        }),  
        stroke: new ol.style.Stroke({  
            color: '#000',  
            width: 3.5  
        })  
    });  
    var selectText = new ol.style.Text({  
        textAlign: 'center',  
        scale: 1.4,  
        fill: new ol.style.Fill({  
            color: '#fff'  
        }),  
        stroke: new ol.style.Stroke({  
            color: '#3399CC',  
            width: 3.5  
        })  
    });  
  
    var positionFeatureStyle = new ol.style.Style({  
        image: new ol.style.Circle({  
            radius: 6,  
            fill: new ol.style.Fill({  
                color: [0, 0, 0, 1]  
            }),  
            stroke: new ol.style.Stroke({  
                color: [255, 255, 255, 1],  
                width: 2  
            })  
        })  
    });  
  
    var accuracyFeatureStyle = new ol.style.Style({  
        fill: new ol.style.Fill({
```

```

        color: [255, 255, 255, 0.3]
    }),
    stroke: new ol.style.Stroke({
        color: [0, 0, 0, 0.5],
        width: 1
    }),
    zIndex: -1
});
var markerFeatureStyle = new ol.style.Style({
    image: new ol.style.Icon({
        anchor: [0.5, 1],
        opacity: 1,
        scale: 0.3,
        src: 'pix/flagmarker.png'
    })
});

```

El fragmento de código anterior, hace referencia a la función *init_map()* que se encarga que establecer lo necesario para la visualización del mapa. En primer lugar, se indica el proveedor del mapa que se utilizará, en este caso “*openstreetmap*” de Open Layers. A continuación, se crean variables de estilo para una correcta visualización con la ayuda de las librerías de Open Layers accesibles mediante el objeto “*ol*”. Una vez creados los estilos se continua con la selección de capas como se puede ver en el siguiente fragmento de código JavaScript.

```

/*-----Layers-----
-----*/
var layers = [];
var geoJSONFormat = new ol.format.GeoJSON();
drawSource = new ol.source.Vector({
    projection: 'EPSG:3857'//Sistema de referencia de
coordenadas para uso correcto con las capas
});
var vectorCustomLayer;
vectorCustomLayer = new ol.layer.Vector({
    source: drawSource,
});
var aeriallayer = new ol.layer.Tile({
    visible: false,
    source: new ol.source.BingMaps({
        key:
'AmC3DXdnK5sXC_Yp_pOLqssFSap1BbvN68jnwKTEM3CSn2t6G5PGTbYN3wzxE5BR',
        imagerySet: 'AerialWithLabels',
        maxZoom: 19
// use maxZoom 19 to see stretched tiles instead of the
BingMaps
// "no photos at this zoom level" tiles
// maxZoom: 19
    })
});
aeriallayer.set("name", "aerialview");
var roadlayer = new ol.layer.Tile({
    source: new ol.source.OSM()
});

```

```

roadlayer.set("name", "roadview");
var layergroup = new ol.layer.Group({ layers: [aeriallayer,
roadlayer] });
var view = new ol.View({
  center: [-474521.071594, 4940889.508354],
  zoom: 6,
  minZoom: 2
});

```

Se crea un vector de capas en el que se almacenarán las capas necesarias para el funcionamiento de la aplicación. A continuación, se indican las propiedades que tomarán las capas como es el sistema de referencia utilizado, en este caso EPSG:3857.

Una vez creada las capas en este caso *aeriallayer* y *roadlayer* se incluyen en una variable de grupo de capas que permitirá su visualización.

En la parte inferior de la aplicación se muestran diferentes botones para realizar diferentes tipos de acciones, a continuación, se verán cada uno de ellos de forma detallada.

CAPAS

El botón capas permite la selección de las diferentes capas que se desean visualizar. Para ello se genera una ventana emergente que permite al usuario seleccionar las capas en cualquier momento.

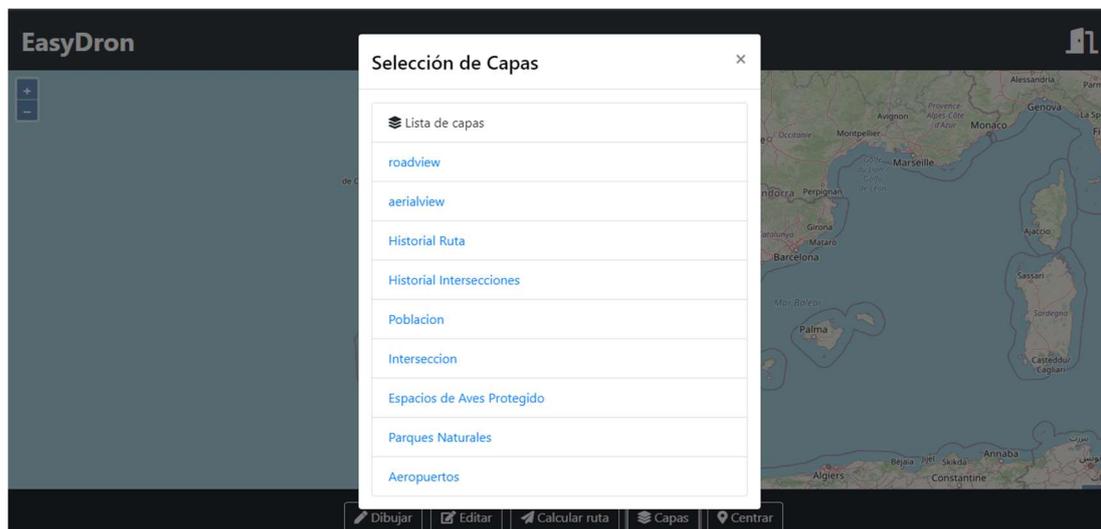


Figura 31: Selección de capas EasyDron.

Como se puede ver en la Figura 31 se muestra una lista de capas que el usuario puede seleccionar solamente pinchando en ellas y de esta forma se mostrarán sobre el mapa. Las capas disponibles son: *Roadview* y *Aerialview* que hacen referencia a los dos tipos

de visualización del mapa, por otra parte, están las capas que se superponen al mapa: Población, Intersección, Espacio de aves protegido, Parques naturales, historial de rutas del usuario y de intersecciones y Aeropuertos algunas de estas solo están disponibles tras realizar la comprobación de una ruta como es la capa Intersección.

Para que esto sea posible en primer lugar es necesario crear un modal en HTML que sea llamado por el botón “Capas”

```
<!--modal para la generación de ventana emergente para permitir la
selección de las capas para su visualización-->
<div class="modal fade" id="capas">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title"
style="color:black">Selección de Capas</h4>
        <button type="button" class="close" data-
dismiss="modal" aria-hidden="true">&times;</button>
      </div>
      <div class="modal-body">
        <ul class="list-group" id="layerslist">
          <li class="list-group-item" id="baseLayer"><span
class="fa fa-layer-group"></span> Lista de capas</li>
        </ul>
      </div>
    </div>
  </div>
</div>
```

Este modal crea una ventana emergente que se encuentra oculta hasta que se pulsa el botón “Capas”, para que se despliegue la ventana es necesario asignar un id, en este caso id=capas. En cuanto al botón es necesario incluir la referencia la id=capas para que pueda acceder a la ventana emergente junto con data-toggle=”modal” para que la despliegue cuando se pulse el botón.

```
<!--botón que llama abre un popup que permite mostrar y seleccionar
las diferentes capas disponibles para visualizarlas-->
<form class="form-inline my-2 my-lg-0 ">
  <a href="#capas" data-toggle="modal" target="capas"
class="btn btn-outline-light my-2 my-sm-0" style="margin-left:
10px" role="button" aria-pressed="true"><span class="fa fa-
layer-group"></span> Capas</a>
</form>
```

Para que la ventana emergente muestre la lista de las capas disponibles en primer lugar es necesario crearlas, para crear las capas que se superponen al mapa y

muestran información al usuario al respecto de las zonas de limitación al vuelo es necesario el siguiente código.

```
InterseccionSource = new ol.source.Vector({
    projection: 'EPSG:3857'
});
var InterseccionLayer = new ol.layer.Vector({
    visible: false,
    source: InterseccionSource,
    style: new ol.style.Style({
        fill: new ol.style.Fill({
            color: '#F70808'
        }),
        stroke: new ol.style.Stroke({
            color: '#F70808',
            width: 4
        })
    })
});

var avesLayer = new ol.layer.Tile({
    visible: false,
    source: new ol.source.TileWMS({
        url: '/geoserver/wms',
        params: {'LAYERS': 'EasyDron:ZEPA', 'TILED': true,
'TRANSSPARENT' : true},
        serverType: 'geoserver',
        // layer have transparency, so do not fade tiles:
        transition: 0
    })
});

var PoblacionLayer = new ol.layer.Tile({
    visible: false,
    source: new ol.source.TileWMS({
        url: '/geoserver/wms',
        params: {'LAYERS': 'EasyDron:Poblacion', 'TILED': true,
'TRANSSPARENT' : true},
        serverType: 'geoserver',
        // layer have transparency, so do not fade tiles:
        transition: 0
    })
});

var parqueLayer = new ol.layer.Tile({
    visible: false,
    source: new ol.source.TileWMS({
        url: '/geoserver/wms',
        params: {'LAYERS': 'EasyDron:ParquesNaturales', 'TILED':
true, 'TRANSPARENT' : true},
        serverType: 'geoserver',
        // layer have transparency, so do not fade tiles:
```

```

        transition: 0
    })
})
var aeroLayer = new ol.layer.Tile({
    visible: false,
    source: new ol.source.TileWMS({
        url: '/geoserver/wms',
        params: {'LAYERS': 'EasyDron:Aeropuertos', 'TILED':
true, 'TRANSPARENT' : true},
        serverType: 'geoserver',
        // layer have transparency, so do not fade tiles:
        transition: 0
    })
})
})

```

Para crear la nueva capa es necesario un nuevo objeto para mostrar la capa en forma de imagen. A continuación, se indica la fuente de los datos y como acceder a ella mediante la url, los parámetros y el tipo de servidor. Una vez creada cada una de las capas se añade a la variable *layers* para crear un objeto *map* en el que poder mostrar las capas

```

layers = [layergroup,
userPosition,vectorCustomLayer,aeroLayer,parqueLayer,avesLayer,Inter
seccionLayer,PoblacionLayer,HistorialInterseccionLayer,
HistorialRutaLayer];
// New Custom zoom.
var zoom = new ol.control.Zoom({ target: "navigation",
className: "custom-zoom" });
map = new ol.Map({
    layers: layers,
    controls: ol.control.defaults({rotate: true, attribution:
true}),
    //interactions: drag_rotate,
    target: 'map',
    view: view
    /*loadTilesWhileAnimating: true,
    loadTilesWhileInteracting: true*/
});

```

Finalmente queda añadir las capas a una lista para poder mostrarlas y seleccionarlas en la ventana emergente, para ello se realiza el siguiente paso con cada una de las capas:

```

aeroLayer.set("name", "Aeropuertos");
add_layer_to_list(aeroLayer);

```

La función *add_layer_to_list()* se encarga de dar formato a la información para poder mostrarla en la ventana emergente y al mismo tiempo realizar la selección o de selección de la capa.

```

function add_layer_to_list(layer) {
  var item = $('<li class="list-group-item">', {
    "data-icon": "check",
    "class": layer.getVisible() ? "checked" : "unchecked"
  })
  .append($('

```

De esta forma se da la posibilidad al usuario de gestionar la información que desea ver en pantalla como se puede ver en el siguiente ejemplo en la Figura 32 que se muestran las Zonas de Especial Protección de Aves (azul) y los Parques Naturales (verde).

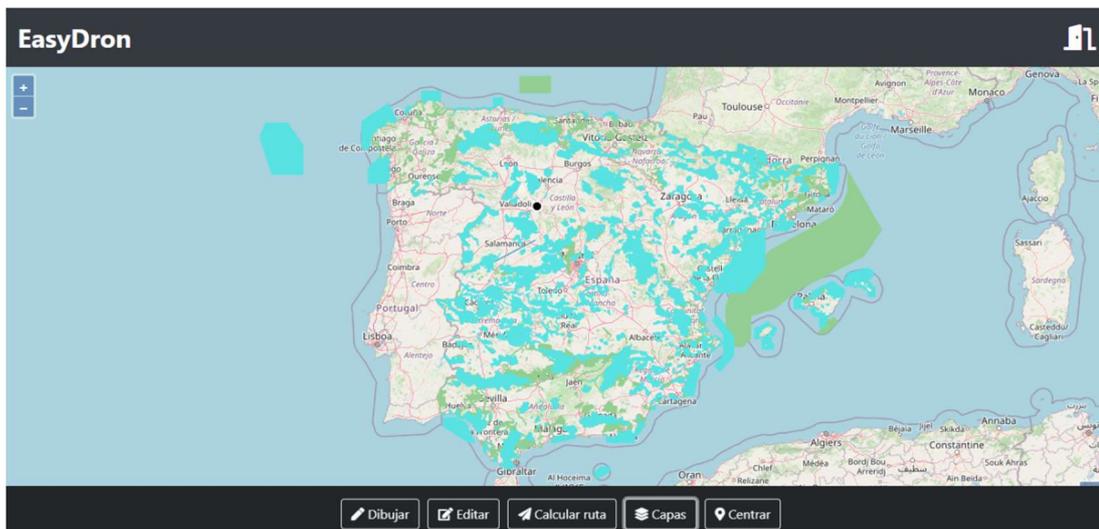


Figura 32: Visualización de capas de ZEPAs y Parques Naturales.

CENTRAR

El botón centrar proporciona la opción de centrar y hacer zoom a la ubicación del usuario, de tal forma que sin tener en cuenta el punto donde se encuentre situada la pantalla lleve la vista a la ubicación con un zoom que permita ver zona con gran precisión como vemos en la Figura 33.

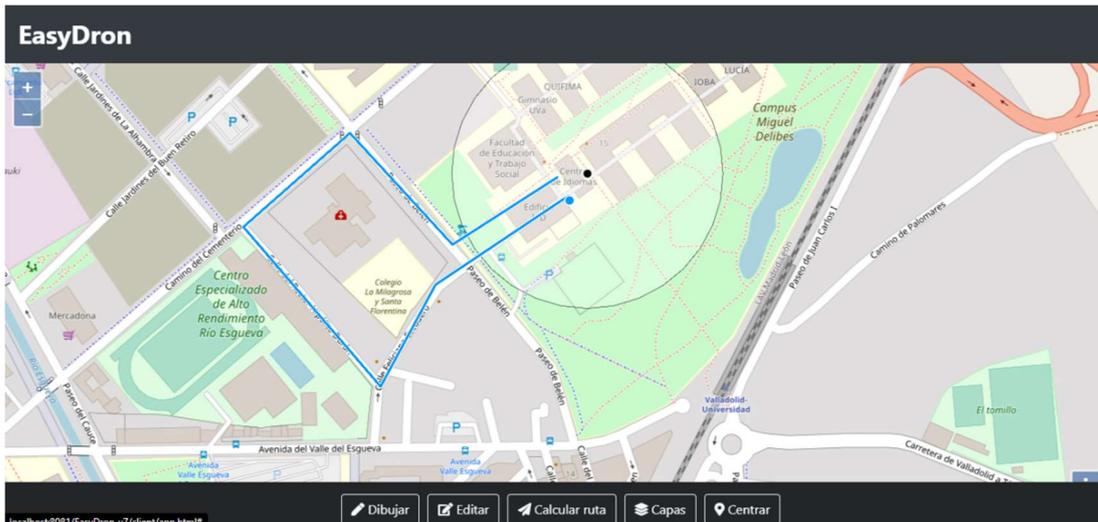


Figura 33: Resultado de centrar

DIBUJAR

La opción de dibujar permite al usuario realizar un trazado por el que se piensa volar el dron, esta ruta puede ser un trazo de una única línea o varias en función de los intereses del usuario.

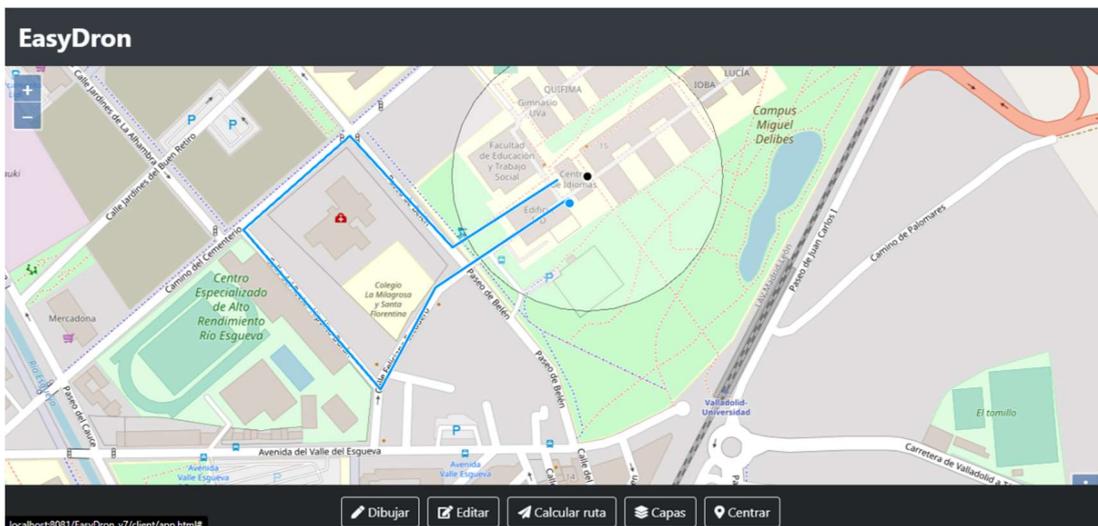


Figura 34: Dibujo de la propuesta de ruta con el dron

Como se puede ver en la Figura 34 la ruta dibujada por el usuario aparece en color azul pudiendo tomar esta la forma deseada por el usuario. Para conseguir este funcionamiento es necesario la creación de una función que genere un variable para almacenar una colección de *features* del tipo *collection* de Open Layers.

```
function dibujar () {
```

```

var collection = new ol.Collection();

if(dibujo == 0)
{
  if (typeof(draw) !='undefined') {
    map.removeInteraction(draw);
  }

  if (typeof(modify) !='undefined') {
    map.removeInteraction(modify);
  }

  drawSource.clear();
  draw = new ol.interaction.Draw({
    features: collection,
    source: drawSource,
    type: "LineString"
  });
  draw.on("drawend", function(event){
    map.removeInteraction(draw);
    geometria=event.feature;
  });
  map.addInteraction(draw);
}
}

```

Se comprueba que no se ha realizado ningún dibujo de una ruta y en tal caso se procede a realizarlo. En primer lugar, se comprueba si hay una ruta tras la acción de dibujar o editar y si existe se elimina del mapa para realizar una nueva. Esto permite al usuario que pueda dibujar sobre el mapa un trazo del tipo línea. Una vez dibujado se almacena las características geométricas del trazo en la variable geometría con la que posteriormente se realizarán los cálculos pertinentes. Este trazo dibujado quedará plasmado sobre el mapa como se puede ver en la Figura 34.

Esta acción se lleva a cabo gracias al botón calcular que es el encargado de llamar a la función *dibujar()* vista anteriormente.

```

<form class="form-inline my-2 my-lg-0 ">
  <a href="#" id="dibujar" class="btn
  btn-outline-light my-2 my-sm-0" style="margin-left: 10px"
  role="button" aria-pressed="true"><span class="fa fa-pen"></span>
  Dibujar</a>
</form>

```

EDITAR

La función editar es útil cuando realizando la ruta el usuario se ha confundido o simplemente desea modificar el dibujo realizado de la ruta siempre que no se haya procedido a realizar los cálculos. Al pulsar el botón “Editar” que se encarga de llamar

a la función *editar()* se permite la modificación de la ruta. Para generar el botón encargado de este proceso es necesario incluir el siguiente código:

```
<!--botón que llama a la función editar, para editar la ruta
dibujada-->
<form class="form-inline my-2 my-lg-0 ">
  <a href="#" id="editar" class="btn btn-outline-light my-2 my-
sm-0" style="margin-left: 10px" role="button" aria-
pressed="true"><span class="fa fa-edit"></span> Editar</a>
</form>
```

La función *editar()* a la que llama contiene el siguiente código:

```
//FUNCION PARA EDITAR RUTA
function editar() {

  if (typeof(modify) !='undefined') {
    map.removeInteraction(modify);
  }
  var selectInteraction = new ol.interaction.Select({
    condition: ol.events.condition.singleClick,
    toggleCondition: ol.events.condition.shiftKeyOnly
  });
  modify = new ol.interaction.Modify({
    source: drawSource,
    features: selectInteraction.getFeatures()
  });
  modify.on('modifyend',function(event) {
    map.removeInteraction(modify);
  });
  map.addInteraction(modify);
  map.getInteractions().extend([selectInteraction, modify]);
}
```

En primer lugar, se comprueba si se ha realizado alguna modificación en la ruta, si es el caso se elimina del mapa. A continuación, se crea la acción de editar que permite mediante un click de ratón arrastrar un punto de la ruta para modificarla y de esta forma se elimina el recorrido de la ruta anterior para pasar a mostrar la nueva ruta dibujada. Y las nuevas características geométricas serán las que se tengan en cuenta para realizar los cálculos.

CALCULAR

La función de calcular accesible desde el botón “Calcular ruta” es la encargada de realizar el procesamiento de la ruta para comprobar su validez. Es la función que mayor peso tiene dentro de la aplicación ya que es la encargada de tratar con los datos

espaciales propios para realizar consultas al servidor Geoserver para obtener diferente información que procesar hasta obtener el resultado de la validez de la ruta.

Cuando se invoca al botón “Calcular” se informará al usuario de los diferentes casos que pueden ocurrir durante la ejecución:

1. Caso 1: No se ha dibujado una ruta. En este caso se informa al usuario de que no se ha dibujado una ruta por lo que no se puede proceder a realizar la comprobación y la función *calcular()* no realiza ninguna acción (ver Figura 35).

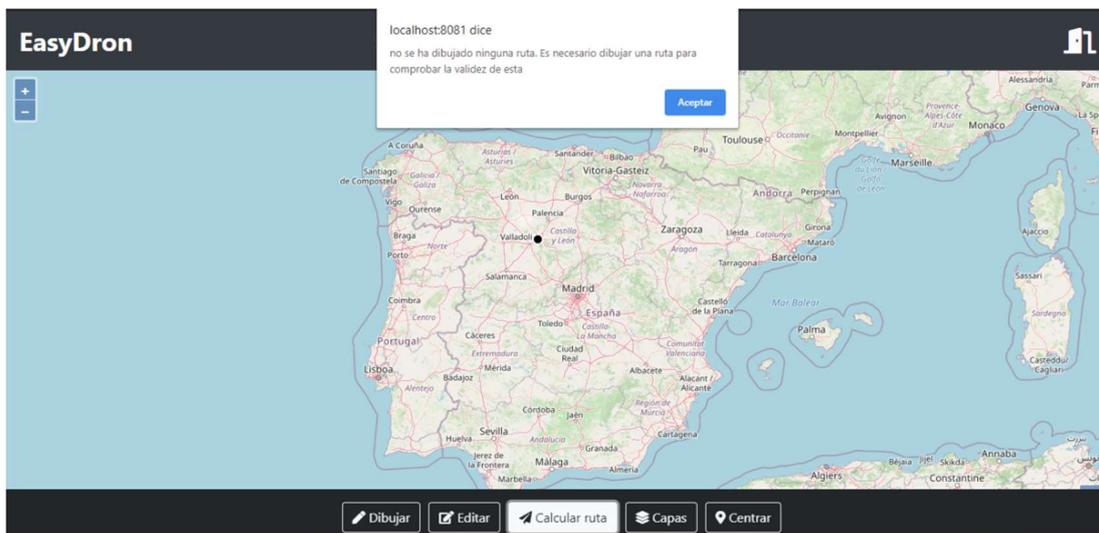


Figura 35: Mensaje informativo de falta de ruta.

2. Caso 2: Se ha dibujado la ruta y se procede a realizar los cálculos que se prolongarán a lo largo de unos segundos, en este momento la función *calcular()* realiza el procesamiento necesario para los datos y muestra al usuario un botón mediante el que acceder a los resultados (ver Figura 36).



Figura 36: Mensaje informativo de inicio de cálculos

Cuando el usuario pulse el botón de Resultados aparecerá un cuadro de dialogo donde se incluirán los resultados bloqueado por un gif de carga hasta que los resultados estén disponibles (ver Figura 37).

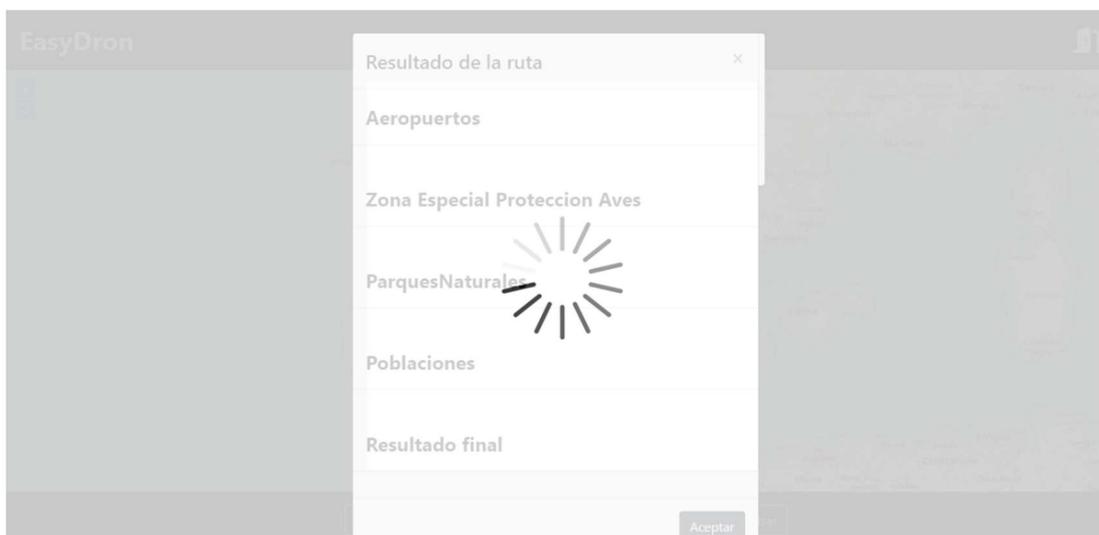


Figura 37: Imagen de carga

CALCULAR()

La función calcular comienza con la comprobación de la existencia de la ruta. En caso de que no se haya dibujado ninguna ruta se informará al usuario para que proceda a dibujarla.

En el caso contrario, es decir, se ha dibujado una ruta comienza el procesamiento para obtener la validez de la ruta.

Para realizar el procesamiento se comienza por la declaración de las distintas variables de opciones. En estas opciones se incluye el sistema de referencia (srsName), la url para acceder a las capas alojadas en Geoserver (featureNS), el nombre del espacio de trabajo (featurePrefix) y la capa sobre la que se realizará la consulta (featureType).

```

else{
    if(calculo==0){
        var
mensajeResultados=document.getElementById('mensajeresultado');
        mensajeResultados.innerHTML = "Los resultados
estarán disponibles en el siguiente enlace";
        //wfs transaccional
        var WFS = new ol.format.WFS();
        var options={
srsName: "EPSG:3857", //proyeccion de openlayers
featureNS: 'http://EasyDron.es',
featurePrefix: 'EasyDron',
featureType: 'rutaDron'
        }
        var options1={
srsName: "EPSG:3857", //proyeccion de openlayers
featureNS: 'http://EasyDron.es',
featurePrefix: 'EasyDron',
featureType: 'interseccion'
        }
        var options2={
srsName: "EPSG:3857", //proyeccion de openlayers
featureNS: '/geoserver/EasyDron/wfs',
featurePrefix: 'EasyDron',
featureType: 'rutaDron'
        }

```

A continuación, se realiza una consulta WFS del tipo *Transaction* para proceder a almacenar la ruta, almacenada en la variable *geometria* en su respectiva capa. Para que la ruta almacene el uid del usuario será necesario incluirlo en la variable *geometria*. Se procede a realizar la consulta mediante el método POST a la dirección donde se encuentra la capa “rutaDron” creada anteriormente en el apartado del servidor.

Esta consulta se realiza sobre la petición WFS de inserción de características (en este caso la geometría de la ruta), para ello es necesario convertir esta consulta a XML mediante la clase *XMLSerializer()* para que pueda ser atendida a través del método POST por el servidor. Una vez completada esta consulta con éxito se almacena la nueva ruta tanto en la capa disponible en Geoserver como en la base de datos Postgres.

```

//añade a la geometria la propiedad de uid del usuario para
almacenar posteriormente
        geometria.setProperties({'uid':uid});
        var node=
WFS.writeTransaction([geometria],null,null,options);

        s = new XMLSerializer();
        str = s.serializeToString(node);
        fetch('/geoserver/EasyDron/wfs',{
method: 'POST',

```



```

        if (b[i]<minimoY)
        {
            ymin=b[i];
            minimoY=ymin;
        }
        if (b[i]>maximoY)
        {
            ymax=b[i];
            maximoY=ymax;
        }
    }
    i++;
}
}

```

De esta forma se obtiene las coordenadas inferiores y superiores de la ruta, es decir, los valores mínimos y máximos de las coordenadas X e Y.

En la siguiente imagen se puede ver una posible ruta en la que se indica en rojo cual sería el encuadre calculado por el algoritmo visto anteriormente (ver Figura 37).

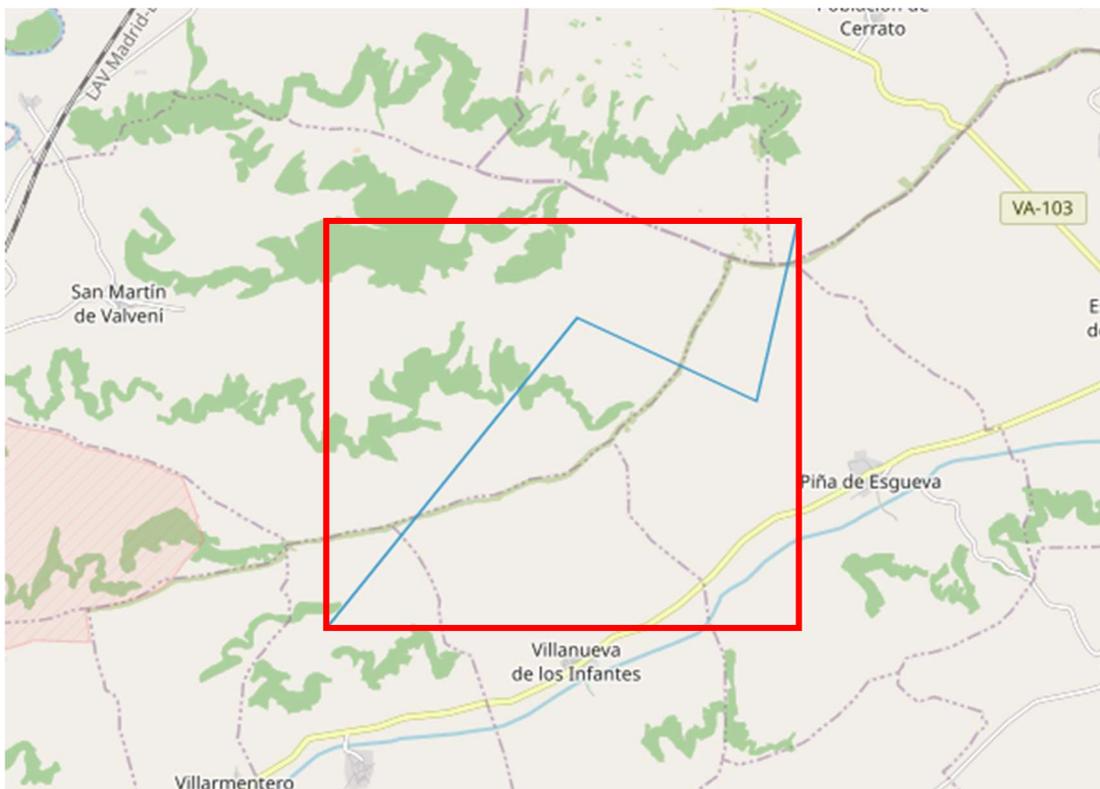


Figura 37: Encuadre de una ruta.

De esta forma los datos para el procesamiento de las distintas capas se limitarán a esta área reduciendo considerablemente la carga de procesamiento.

A continuación, se realiza una consulta WFS de las rutas almacenadas por el usuario para seleccionar el fid de la última realiza por el mismo y obtener la ruta almacenada anteriormente en el formato de Open Layers. Esta consulta tiene formato XML en la que se indica en primer lugar los servicios utilizados, versiones formatos utilizados para realizarla, además de incluir la asignación a las etiquetas de los diferentes servicios que permiten realizar la consulta. Esta consulta sirve para obtener todas las *features* o características disponibles en la capa “rutaDron” almacenadas en el espacio de trabajo “EasyDron” de Geoserver accesible a través de <http://EasyDron.es>

```
//consulta para obtener las rutas del usuario
var rutas_usuario = `

```

```

xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs
http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">
  <wfs:Query typeName="EasyDron:rutaDron">
    <ogc:Filter>
      <ogc:FeatureId fid="{fid_rutausuario}"/>
    </ogc:Filter>
  </wfs:Query>
</wfs:GetFeature> `;

```

Esta última consulta almacenada en “ruta” obtiene la última ruta registrada en la base de datos. A continuación, se realizará el procesamiento de cada uno de los casos mencionados en los apartados anteriores.

Comenzando con el caso de los Parques Naturales y ZEPAs será necesario la comprobación de la existencia de lugares de ambas zonas en el encuadre de la ruta. El motivo de realizar el procesamiento diferenciando cada caso se debe a que el proceso de unión que se realiza mediante la consulta WPS solo se obtiene un resultado exitoso en el caso de que existan valores no nulos para ambas capas.

Para comenzar este procesamiento en primer lugar tenemos que saber si existen ZEPAs y/o parques naturales en el espacio de la ruta, para ello se realizan las siguientes consultas que obtienen las *features* de las capas en el encuadre de la ruta.

```

//comprueba si existen parques naturales en la zona de la ruta
var getParques = `<wfs:GetFeature service="WFS" version="1.1.0"
  xmlns:EasyDron="http://EasyDron.es"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs
    http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">
  <wfs:Query typeName="EasyDron:ParquesNaturales">
    <ogc:Filter>
      <ogc:BBOX>
        <ogc:PropertyName>geom</ogc:PropertyName>
        <Envelope
srsName="http://www.opengis.net/gml/srs/epsg.xml#3857">
          <lowerCorner>${minimoX} ${minimoY}</lowerCorner>
          <upperCorner>${maximoX} ${maximoY}</upperCorner>
        </Envelope>
      </ogc:BBOX>
    </ogc:Filter>
  </wfs:Query>

```

```
</wfs:GetFeature>`;
```

```
var getZEPA = `<wfs:GetFeature service="WFS" version="1.1.0"  
    xmlns:EasyDron="http://EasyDron.es"  
    xmlns:wfs="http://www.opengis.net/wfs"  
    xmlns:ogc="http://www.opengis.net/ogc"  
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
    xsi:schemaLocation="http://www.opengis.net/wfs  
    http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">  
  <wfs:Query typeName="EasyDron:ZEPA">  
    <ogc:Filter>  
      <ogc:BBOX>  
        <ogc:PropertyName>geom</ogc:PropertyName>  
        <Envelope  
          srsName="http://www.opengis.net/gml/srs/epsg.xml#3857">  
            <lowerCorner>${minimoX}${minimoY}</lowerCorner>  
            <upperCorner>${maximoX}${maximoY}</upperCorner>  
          </Envelope>  
        </ogc:BBOX>  
      </ogc:Filter>  
    </wfs:Query>  
</wfs:GetFeature>`;
```

Para estas consultas utiliza las mismas cabeceras y etiquetas que en el caso anterior la diferencia es que para la obtención de las *features* se aplica un filtro a la propiedad con el nombre “geom” que corresponde a la geometría de la *feature*, de tal forma que solo se obtengan aquellas cuya geometría se encuentra dentro del encuadre requerido. El encuadre se indica en las líneas

```
<lowerCorner>${minimoX}${minimoY}</lowerCorner>  
<upperCorner>${maximoX}${maximoY}</upperCorner>
```

Donde las variables $\{\text{minimoX}\}$, $\{\text{minimoY}\}$, $\{\text{maximoX}\}$ y $\{\text{maximoY}\}$ son las coordenadas calculadas anteriormente, en la búsqueda de las coordenadas máximas y mínimas de X e Y.

Para obtener el número de *features* que existen en esta zona es necesario llamar a la función *wpsclient_count()* como en el caso de la ruta pero cambiando el parámetro de entrada por el correspondiente a cada capa. Siendo los parámetros los siguientes:

```
var href='http://localhost:8081/geoserver/EasyDron/wps';  
var prefix = 'feature';  
var prefixBD= 'EasyDron';  
var namespace = 'http://EasyDron.es';  
var projection = ol.proj.get("EPSG:3857");  
var featuretype2 = 'ZEPA';  
var featuretypeParques='ParquesNaturales';
```

En primer lugar, se pasa la dirección de acceso al servicio WPS del espacio de trabajo, en este caso se encuentra en localhost:8081, pero este puede variar en función del punto en el que se realiza la instalación del servidor Geoserver. “Prefix” hace referencia al prefijo de los atributos almacenados de cada *feature* para poder almacenar su valor, “PrefixBD” hace referencia al nombre del espacio de trabajo

donde se encuentran los recursos. La variable “namespace” es la url por la que se acceden a los recursos almacenados en el espacio de trabajo, donde el nombre del recurso se almacena en este caso en “featuretype3”.

```
//Lanza una petición al WPS asíncrona para obtener el número de
parques naturales en el bounding box de la ruta
var cuentaParques = await wpsclient_count(href, numeroParques,
prefixBD, namespace, featuretypeParques, projection);
//obtiene el numero de ZEPAs en el bounding box de la ruta
var cuentaZEPa = await wpsclient_count(href, numeroZEPa, prefixBD,
namespace, featuretype2, projection);
```

Con el número de parques naturales y de ZEPAs en el encuadre de la ruta queda realizar el procesamiento teniendo en cuenta los casos mencionados anteriormente.

1. Caso 1: Existen ZEPAs y parques naturales

En este caso se realiza la unión de las dos capas y de esta forma unificar el resultado y realizar un procesamiento más eficiente que tratar las capas por separado. El proceso de unión es el siguiente:

```
if(cuentaZEPa!=0 && cuentaParques!=0) {

//consulta WPS para para la union de los parques naturales con las
ZEPAS en el bounding box de la ruta
var UnionWPS=wps:Execute version="1.0.0" service="WPS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.opengis.net/wps/1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:wcs="http://www.opengis.net/wcs/1.1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsAll.xsd">
<ows:Identifier>gs:UnionFeatureCollection</ows:Identifier>
  <wps:DataInputs>
    <wps:Input>
      <ows:Identifier>first</ows:Identifier>
      <wps:Reference mimeType="text/xml; subtype=wfs-
collection/1.1" xlink:href="{baseurl}/geoserver/wfs" method="POST">
      <wps:Body><![CDATA[<wfs:GetFeature          service="WFS"
version="1.1.0"
xmlns:EasyDron="http://EasyDron.es"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/wfs
http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">
<wfs:Query typeName="EasyDron:ZEPa">
      <ogc:Filter>
```

```

        <ogc:BBOX>
          <ogc:PropertyName>geom</ogc:PropertyName>
        </ogc:BBOX>
      </ogc:Filter>
    </wfs:Query>
  </wfs:GetFeature>]]></wps:Body>
  </wps:Reference>
  </wps:Input>
  <ows:Identifier>second</ows:Identifier>
  <wps:Reference mimeType="text/xml; subtype=wfs-
collection/1.1" xlink:href="{baseurl}/geoserver/wfs" method="POST">
  <wps:Body><![CDATA[<wfs:GetFeature service="WFS"
version="1.1.0"
  xmlns:EasyDron="http://EasyDron.es"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
  xsi:schemaLocation="http://www.opengis.net/wfs
http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">
  <wfs:Query typeName="EasyDron:ParquesNaturales">
    <ogc:Filter>
      <ogc:BBOX>
        <ogc:PropertyName>geom</ogc:PropertyName>
      </ogc:Filter>
    </ogc:Filter>
  </wfs:Query>
  </wfs:GetFeature>]]></wps:Body>
  </wps:Reference>
  </wps:Input>
</wps>DataInputs>
<wps:ResponseForm>
  <wps:RawDataOutput mimeType="text/xml;
subtype=wfs-collection/1.1">
    <ows:Identifier>result</ows:Identifier>
  </wps:RawDataOutput>
</wps:ResponseForm>
</wps:Execute>`;

```

Este proceso WPS de unión está formado por dos entradas, la primera entrada es un proceso WFS de obtención de las *features* de la capa ZEPa cuyas geometrías se encuentran en el encuadre de la ruta, la segunda entrada es otro proceso WFS igual que el anterior, pero con la capa de ParquesNaturales. El resultado de este proceso es una colección de *features* cuyos atributos quedan nombrados con el nombre original

del atributo precedido del nombre de la capa de la primera entrada junto con un guion bajo, es decir, ZEPA_.

Esta consulta es necesaria para a continuación realizar una consulta de intersección para saber si la ruta pasa por alguna de estas zonas. Este proceso de intersección es un proceso WPS que cuenta con dos entradas, la primera entrada es el proceso anterior que se incluye mediante la variable en la que se guarda la consulta de unión en forma de cadena y la segunda entrada es un proceso WFS para obtener los atributos y valores de la ruta dibuja para su consulta:

```
// consulta WPS de la interseccion de la union de parques naturales
y ZEPA con la ruta
var InterseccionConAvesParques=`<?xml version="1.0" encoding="UTF-
8"?><wps:Execute version="1.0.0" service="WPS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.opengis.net/wps/1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:wcs="http://www.opengis.net/wcs/1.1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsAll.xsd">
<ows:Identifier>gs:IntersectionFeatureCollection</ows:Identifier>
  <wps>DataInputs>
    <wps:Input>
      <ows:Identifier>first feature
collection</ows:Identifier>
      <wps:Reference mimeType="text/xml"
xlink:href="http://geoserver/wps" method="POST">
        <wps:Body>
          ${UnionWPS}
        </wps:Body>
      </wps:Reference>
    </wps:Input>
    <wps:Input>
      <ows:Identifier>second feature
collection</ows:Identifier>
      <wps:Reference mimeType="text/xml;
subtype=wfs-collection/1.1" xlink:href="{baseurl}/geoserver/wfs"
method="POST">
        <wps:Body>
          <![CDATA[ ${ruta} ]]>
        </wps:Body>
      </wps:Reference>
    </wps:Input>
  </wps>DataInputs>
  <wps:ResponseForm>
    <wps:RawDataOutput mimeType="text/xml;
subtype=wfs-collection/1.1">
      <ows:Identifier>result</ows:Identifier>
    </wps:RawDataOutput>
```

```

</wps:ResponseForm>
</wps:Execute>`;

```

Almacenando en la variable “InterseccionConAvesParques” la consulta de intersección completa junto con la unión se pasa como parámetro a la función *wpsclient_featurecollection()* para obtener las features resultado de la intersección para poder tratarlas y obtener unos resultados válidos para mostrar al usuario.

```

//obtiene las features de la intersección de la unión de parques
naturales y ZEPA con la ruta
var interseccionConAP = await wpsclient_featurecollection(href,
InterseccionConAvesParques, prefix, namespace, featuretype2,
projection);

```

La función *wpsclient_featurecollection()* permite realizar una consulta con el método POST con los parámetros de entrada necesarios para realizar las consultas necesarias a los diferentes recursos. Además, la función trata los datos para obtener los atributos y sus valores en un formato válido para tratarlos mediante JavaScript y poder mostrarlos posteriormente al usuario mediante la aplicación.

```

function wpsclient_featurecollection(href, wpsbody, prefix,
namespace, featuretype, projection){
    var SRScode= projection.getCode().substring(5);
    var WPSSRSname = "http://www.opengis.net/gml/srs/epsg.xml#" +
SRScode;

return fetch(href, {
    method: "POST",
    headers: {
        "Content-Type": "application/xml"
    },
    body: wpsbody
}).then(function(response){
    return response.text();
}).then(function(gml){
    var doc = ol.xml.parse(gml);
    var colls =
doc.getElementsByTagName("wfs:FeatureCollection");
    var coll = colls[0];
    var options={
        srsName: projection.getCode(),
        featureNS: namespace
        featurePrefix: prefix,
        featureType: featuretype
    }
    // Register the alias for the SRS.

```

```

proj4.defs(WPSSRSname, projection);
var wfsformat = new ol.format.GML(options);
// Parche para leer el GML del WPS de Geoserver.
wfsformat.readFeaturesJP =function(node) {
var features = [];
if (node) {var childNodes = node.childNodes;
for(var i=0; i<childNodes.length; i++) {
if (childNodes[i].localName == 'featureMember') {
var featureNode=childNodes[i].childNodes[0];
var feature=this.readFeatures(childNodes[i]);
var feature1 = this.readFeatures(childNodes[i]);
if(feature) {
features.push(feature);
}
}
}
return features;
};
var features = wfsformat.readFeaturesJP(coll);
return Promise.resolve(features);
});
}

```

Tras realizar un cambio de formato para poder leer las *features* se comprueba que existen y tienen un valor asociado y en tal caso la función las devuelve al vector “interseccionConAP” donde se almacenan para su posterior tratamiento.

A continuación será necesario incluir al resultado obtenido anteriormente datos de la ruta y del usuario como son “idRuta” el fid de la ruta que se está calculando y el uid del usuario que realiza la ruta. Se realiza el proceso de inserción como el realizado en el caso de la ruta, pero a diferencia que se realiza una vez por cada *feature* almacenada en el vector.

```

//calculo el número de features de intersección que hay con la ruta
var numeroFeatures=interseccionConAP.length;
for(var i=0; i<numeroFeatures; i++){
//Añade una nueva feature para tener un identificador de la
ruta con la que intersecciona
//Envió cada una de las intersecciones a geoserver
interseccionConAP[i].set('idRuta', fid_rutausuario);
interseccionConAP[i].set('uid', uid);
var nodel=
WFS.writeTransaction([interseccionConAP[i]],null,null,options1);

s = new XMLSerializer();
str = s.serializeToString(nodel);
fetch('/geoserver/EasyDron/wfs',{
method: 'POST',
body: str
}).then(function (nodel){
return nodel.text();
}).then(function(res){
var resultado1=WFS.readTransactionResponse(res);
});
}

```

De esta forma quedará actualizada tanto la capa en Geoserver como la base de datos Postgres. Con los datos registrados, es el momento de procesarlos para convertirlos a un formato que pueda mostrarse al usuario ya que en este instante están almacenados como *features* de Open Layers, para ello se obtienen todos los valores que resultan de interés en cada caso, en este concreto se obtiene el nombre de las ZEPAs y su responsable y el nombre de los Parques Naturales y sus responsables. Este valor se almacena en una variable, posteriormente se comprueba que esta variable tiene un valor y en tal caso se almacena en un vector. Esta comprobación se realiza ya que al almacenar al realizar la intersección se crea una nueva colección de *features* en las que se incluyen los campos de los parques naturales y las ZEPAs, lo que da lugar a que en el caso que se almacena en un parque natural los campos de ZEPAs queden vacíos en esa colección.

```
for (var i=0; i<numeroFeatures; i++){

    valorParques= interseccionConAP[i].values_['ZEPA_sitename']
    if (valorParques!=undefined)
        vectorParques.push(valorParques);
    valorParquesR= interseccionConAP[i].values_['ZEPA_ccaa_n_enp']
    if(valorParquesR!=undefined)
        vectorRParques.push(valorParques);
    valorAves= interseccionConAP[i].values_['ZEPA_site_name']
    if (valorAves!=undefined)
        vectorAves.push(valorAves);
    valorAvesR= interseccionConAP[i].values_['ZEPA_ac']
    if(valorAvesR!=undefined)
        vectorRAves.push(valorAvesR)

}
```

2. Caso 2: Solo existen ZEPAs

En este caso que no existen parques naturales en la zona, el procesado se realiza únicamente con las ZEPAs para ello se realiza una consulta al servidor para obtener la intersección de la ruta con las ZEPAs que se encuentran en el encuadre de esta.

```
else if(cuentaZEPA!=0 && cuentaParques==0){

//consulta WPS de la interseccion de las zonas de especial
proteccion de aves con las rutas
var InterseccionConZEPA=`<?xml version="1.0" encoding="UTF-
8"?><wps:Execute version="1.0.0" service="WPS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.opengis.net/wps/1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
```

```

xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:wcs="http://www.opengis.net/wcs/1.1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsAll.xsd">
<ows:Identifier>gs:IntersectionFeatureCollection</ows:Identifer>
  <wps:DataInputs>
    <wps:Input>
      <ows:Identifier>first feature collection</ows:Identifier>
      <wps:Reference mimeType="text/xml"
xlink:href="{baseurl}/geoserver/wps" method="POST">
        <wps:Body>
          <![CDATA[<wfs:GetFeature service="WFS" version="1.1.0"
outputFormat="GML2"
  xmlns:EasyDron="http://EasyDron.es"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs
http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">
    <wfs:Query typeName="EasyDron:ZEPA">
      <ogc:Filter>
        <ogc:BBOX>
          <ogc:PropertyName>geom</ogc:PropertyName>
          <Envelope
srsName="http://www.opengis.net/gml/srs/epsg.xml#3857">
            <lowerCorner>${minimoX} ${minimoY}</lowerCorner>
            <upperCorner>${maximoX} ${maximoY}</upperCorner>
            </Envelope>
          </ogc:BBOX>
        </ogc:Filter>
      </wfs:Query>
    </wfs:GetFeature>]]>
        </wps:Body>
      </wps:Reference>
    </wps:Input>
    <wps:Input>
      <ows:Identifier>second feature collection</ows:Identifier>
      <wps:Reference mimeType="text/xml; subtype=wfs-collection/1.1"
xlink:href="{baseurl}/geoserver/wfs" method="POST">
        <wps:Body>
          <![CDATA[ ${ruta} ]]>
        </wps:Body>
      </wps:Reference>
    </wps:Input>
  </wps:DataInputs>
  <wps:ResponseForm>
    <wps:RawDataOutput mimeType="text/xml; subtype=wfs-
collection/1.1">
      <ows:Identifier>result</ows:Identifier>
    </wps:RawDataOutput>
  </wps:ResponseForm>
</wps:Execute>`;

```

La primera entrada de la consulta se trata de una consulta WFS para obtener las *features* que se encuentran en el encuadre de la ruta y la segunda entrada es la ruta que

ha dibujado el usuario de esta forma tras realizar la consulta al servidor se obtiene si hay o no hay puntos en los que interseca con zonas de limitación al vuelo.

Se realiza de nuevo una consulta mediante el método POST para almacenar este resultado en la parte del servidor y se procesan los datos para que puedan ser mostrados al usuario. En este caso no es necesario realizar la comprobación como en el caso anterior ya que ahora tendrá valores en todos los casos.

```
//almacena los datos de ZEPA con los que interseca
for (var i=0; i<numeroFeatures; i++){
    valorAves= interseccionZEPA[i].values_['ZEPA_site_name']
    vectorAves.push(valorAves);
    valorAvesR= interseccionZEPA[i].values_['ZEPA_ac']
    vectorRAves.push(valorAvesR);
}
}
```

3. Caso 3: Solo hay parques naturales

En este caso la forma de proceder es igual que en el caso 2, únicamente cambia el nombre de la capa a la que se accede como primera entrada, siendo este caso la capa ParquesNaturales.

```
var InterseccionConParques=`<?xml version="1.0" encoding="UTF-8"?><wps:Execute version="1.0.0" service="WPS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.opengis.net/wps/1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:wcs="http://www.opengis.net/wcs/1.1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsAll.xsd">
<ows:Identifier>gs:InterseccionFeatureCollection</ows:Identifier>
  <wps:DataInputs>
    <wps:Input>
      <ows:Identifier>first feature collection</ows:Identifier>
      <wps:Reference mimeType="text/xml"
xlink:href="{baseurl}/geoserver/wps" method="POST">
      <wps:Body>
        <![CDATA[<wfs:GetFeature service="WFS" version="1.1.0"
outputFormat="GML2"
  xmlns:EasyDron="http://EasyDron.es"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs
http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">
    <wfs:Query typeName="EasyDron:ParquesNaturales">
      <ogc:Filter>
        <ogc:BBOX>
          <ogc:PropertyName>geom</ogc:PropertyName>
```

```

        <Envelope
srsName="http://www.opengis.net/gml/srs/epsg.xml#3857">
            <lowerCorner>${minimoX} ${minimoY}</lowerCorner>
            <upperCorner>${maximoX} ${maximoY}</upperCorner>
            </Envelope>                                </ogc:BBOX>
        </ogc:Filter>
    </wfs:Query>
</wfs:GetFeature>]]>
</wps:Body>
</wps:Reference>
</wps:Input>
<wps:Input>
    <ows:Identifier>second feature collection</ows:Identifier>
    <wps:Reference mimeType="text/xml; subtype=wfs-collection/1.1"
xlink:href="\${baseurl}/geoserver/wfs" method="POST">
        <wps:Body>
        <![CDATA[\${ruta}]]>
        </wps:Body>
    </wps:Reference>
</wps:Input>
</wps:DataInputs>
<wps:ResponseForm>
    <wps:RawDataOutput mimeType="text/xml; subtype=wfs-
collection/1.1">
        <ows:Identifier>result</ows:Identifier>
    </wps:RawDataOutput>
</wps:ResponseForm>
</wps:Execute>`;

```

Con este procedimiento que se ha descrito se ha calculado el paso de la ruta del dron por una Zona de Especial Protección de Aves y/o parques naturales, obteniendo los resultados necesarios para informar al usuario de las limitaciones al vuelo de grado medio.

A continuación, se procede a realizar los cálculos para obtener la validez de la ruta por zonas donde la limitación al vuelo es de un grado alto, estas zonas son las de aeropuertos en las que se extiende la zona hasta un radio de 8km del aeropuerto. El primer paso en este procedimiento es obtener la circunferencia de 8km de radio a partir de la geometría del aeropuerto. Para ello se utiliza la consulta “BufferFeatureCollection” de WPS que permite el cálculo de una nueva geometría a partir de la seleccionada, ampliando el área que ocupa. En este caso se realiza un buffer circular de de 8km de radio. La geometría seleccionada mediante el filtro en el que se indica que se realiza sobre la propiedad “tip_area = 1” hace referencia a la geometría de aeródromo del aeropuerto ya que en un aeropuerto podemos encontrar diversas geométricas como puede ser la pista de aterrizaje, pista de rodaje...

Además, solo se toman las geometrías que se encuentran en el encuadre de la ruta. En cuanto a la distancia deseada se indica mediante el parámetro *distance*, se trata de una distancia cartesiana o plana, como el sistema de coordenadas empleado es proyectado emplearemos la distancia en metros, en este caso 8000.//consulta WPS para obtener el buffer de 8km de los aeropuertos

```

var BufferWPS=`<wps:Execute version="1.0.0" service="WPS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.opengis.net/wps/1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"

```

```

xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:wcs="http://www.opengis.net/wcs/1.1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsAll.xsd">
<ows:Identifier>gs:BufferFeatureCollection</ows:Identifier>
  <wps>DataInputs>
    <wps:Input>
      <ows:Identifier>features</ows:Identifier>
      <wps:Reference mimeType="text/xml"
xlink:href="http://geoserver/wfs" method="POST">
        <wps:Body>
          <wfs:GetFeature service="WFS" version="1.0.0"
outputFormat="GML2" xmlns:EasyDron="http://EasyDron.es">
            <wfs:Query typeName="EasyDron:Aeropuertos">
              <ogc:Filter>
                <ogc:And>
                  <ogc:PropertyIsEqualTo>
                    <ogc:PropertyName>tip_area</ogc:PropertyName>
                    <ogc:Literal>1</ogc:Literal>
                  </ogc:PropertyIsEqualTo>
                  <ogc:BBOX>
                    <ogc:PropertyName>geom</ogc:PropertyName>
                  </ogc:BBOX>
                </ogc:And>
              </wfs:Query>
            </wfs:GetFeature>
          </wps:Body>
        </wps:Reference>
      </wps:Input>
    </wps>DataInputs>
    <wps:ResponseForm>
      <wps:RawDataOutput mimeType="text/xml; subtype=wfs-
collection/1.0">
        <ows:Identifier>result</ows:Identifier>
      </wps:RawDataOutput>
    </wps:ResponseForm>
  </wps:Execute>`;

```

De esta forma se consigue los puntos de la ruta que intersecan con el espacio aéreo restringido por la proximidad a un aeropuerto en menos de 8km. Se procesa la consulta

mediante la consulta HTTP con el método POST para incluirla en la capa intersección en caso de que el resultado sea positivo.

Por último, queda realizar la comprobación de la validez de la ruta con las poblaciones, esta zona es conflictiva en todo caso para drones mayores de 250gr, siempre que no se disponga de permiso de la autoridad competente, en caso de que el peso del dron sea menor de 250gr es posible que pueda realizar algún vuelo, pero sujeto a limitaciones. En este caso se realiza la intersección de todas las poblaciones que se encuentran en las proximidades de la ruta, pero sin llevar a cabo ningún tipo de manipulación de los datos. Para ello se realiza la siguiente consulta que proporciona el resultado buscado tras enviarlo como petición al servidor Geoserver.

```
var InterseccionConZEPA=`<?xml version="1.0" encoding="UTF-8"?><wps:Execute version="1.0.0" service="WPS"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.opengis.net/wps/1.0.0"
xmlns:wfs="http://www.opengis.net/wfs"
xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:gml="http://www.opengis.net/gml"
xmlns:ogc="http://www.opengis.net/ogc"
xmlns:wcs="http://www.opengis.net/wcs/1.1.1"
xmlns:xlink="http://www.w3.org/1999/xlink"
xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
http://schemas.opengis.net/wps/1.0.0/wpsAll.xsd">
<ows:Identifier>gs:IntersectionFeatureCollection</ows:Identifer>
  <wps>DataInputs>
    <wps:Input>
      <ows:Identifier>first feature collection</ows:Identifier>
      <wps:Reference mimeType="text/xml"
xlink:href="{baseurl}/geoserver/wps" method="POST">
        <wps:Body>
          <![CDATA[<wfs:GetFeature service="WFS" version="1.1.0"
outputFormat="GML2"
  xmlns:EasyDron="http://EasyDron.es"
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns:ogc="http://www.opengis.net/ogc"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs
http://schemas.opengis.net/wfs/1.1.0/wfs.xsd">
    <wfs:Query typeName="EasyDron:Poblacion">
      <ogc:Filter>
        <ogc:BBOX>
          <ogc:PropertyName>geom</ogc:PropertyName>
          <Envelope
srsName="http://www.opengis.net/gml/srs/epsg.xml#3857">
            <lowerCorner>${minimoX} ${minimoY}</lowerCorner>
            <upperCorner>${maximoX} ${maximoY}</upperCorner>
          </Envelope>
        </ogc:BBOX>
      </ogc:Filter>
    </wfs:Query>
  </wfs:GetFeature>]]>
        </wps:Body>
```

```

</wps:Reference>
</wps:Input>
<wps:Input>
<ows:Identifier>second feature collection</ows:Identifier>
<wps:Reference mimeType="text/xml; subtype=wfs-collection/1.1"
xlink:href="{baseurl}/geoserver/wfs" method="POST">
  <wps:Body>
  <![CDATA[{ruta}]]>
  </wps:Body>
</wps:Reference>
</wps:Input>
</wps>DataInputs>
<wps:ResponseForm>
<wps:RawDataOutput mimeType="text/xml; subtype=wfs-
collection/1.1">
  <ows:Identifier>result</ows:Identifier>
</wps:RawDataOutput>
</wps:ResponseForm>
</wps:Execute>`;

```

Tras realizar todas las consultas necesarias a la parte del servidor solamente queda terminar de procesar los datos para poder mostrarlos al usuario. Estos datos se almacenan en forma de cadenas para que puedan ser incluidos en contenido HTML que sea visible para el usuario. Este tratamiento y muestra de datos se verán en el siguiente apartado de resultados.

RESULTADO

Esta funcionalidad es accesible al usuario a través del botón “Resultado” ubicado en la ventana emergente de información tras iniciar el cálculo de la ruta, lo que le permitirá abrir una ventana desplegable en la que se mostrará información relativa al proceso. La información que el usuario puede visualizar es la siguiente:



Figura 39: Resultado negativo



Figura 40: Resultado positivo

Se muestra la información relativa al éxito o fracaso de la ruta. En caso de que la ruta sea exitosa por una zona se muestra un mensaje indicando que no supone ningún problema para el vuelo relativa a la zona especificada (ver Figura 40). Por el contrario, si la ruta se ve afectada por una zona con limitación se indicará en cada apartado la información de la zona afectada (ver Figura 39).

Como se puede ver en la Figura 39 se indica que existe una limitación de vuelo debido al paso de la ruta por el Aeropuerto de Villanubla e indica también el responsable del aeropuerto por si es necesario establecer contacto para solicitar algún tipo de permiso. Por otra parte, se puede ver que en la parte final se muestra un mensaje indicando la validez total de la ruta, en este caso al pasar por una zona limitada el resultado final indica que no es válida para practicar el vuelo. En caso contrario, es decir, que la ruta no pase por una zona limitada el mensaje mostrado sería el de la Figura 40.

Para poder mostrar esta información al usuario es necesario realizar en primer lugar un modal en el documento de la aplicación para que pueda ser abierto en el momento que se pulsa el botón.

```
<!--botón que abre un popup para mostrar los mensajes relativos al
cálculo de la ruta-->
<form class="form-inline my-2 my-lg-0 ">
  <a href="#resultado" data-toggle="modal" target="resultado"
class="btn btn-outline-light my-2 my-sm-0" style="margin-left: 10px"
role="button" aria-pressed="true"><span class="fa fa-clipboard-
check"></span> Resultado</a>
</form>
```

Este botón abre el modal con id=resultado, este modal muestra una pantalla de carga hasta que los resultados están listos para su visualización.

```
<!--modal para la generacion de ventana emergente para informar de
los resultados obtenidos de los cálculos-->
<div class="modal fade" id="resultado">
  <div class="loader"></div>
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h4 class="modal-title" style="color:black">Resultado de
la ruta</h4>
        <button type="button" class="close" data-dismiss="modal"
aria-hidden="true">&times;</button>
      </div>
      <div class="modal-header">
        <h4 class="modal-
title" style="color:black"><strong>Aeropuertos</strong></h4>
      </div>
      <div class="modal-body" id="resultadoAeropuertos">

      </div>
      <div class="modal-header">
        <h4 class="modal-
title" style="color:black"><strong>Zona Especial Proteccion
Aves</strong></h4>
      </div>
      <div class="modal-body" id="resultadoAves">

      </div>
      <div class="modal-header">
        <h4 class="modal-
title" style="color:black"><strong>ParquesNaturales</strong></h4>
      </div>
      <div class="modal-body"
id="resultadoParquesNaturales">

      </div>
      <div class="modal-header">
        <h4 class="modal-
title" style="color:black"><strong>Poblaciones</strong></h4>
      </div>
      <div class="modal-body" id="resultadoPoblacion">
```

```

        </div>
        <div class="modal-header">
            <h4 class="modal-
title" style="color:black"><strong>Resultado final</strong></h4>
        </div>
        <div class="modal-body" id="resultadoFinal">

        </div>
        <div class="modal-footer">
            <button type="button" class="btn btn-dark" data-
dismiss="modal">Aceptar</button>
            <button type="button" class="btn btn-dark" data-
dismiss="modal">Intentar de nuevo</button>
        </div>

```

La ventana emergente posee diferentes secciones, el título en el que se indica el nombre de la ventana junto el botón de cerrar la ventana, el contenido donde se encuentra la información de los resultados y por último el pie donde se alojan los dos botones. En la sección del contenido se distinguen varios apartados, uno por cada zona de estudio para la comprobación de la ruta y uno como información final de la comprobación. Estos apartados están formados por el título en el que se indica el nombre de la zona que se estudia y un contenido que es variable en función del caso en el que se encuentre, gracias a un identificador que permite incluir la información que se tiene de los cálculos realizados.

Para mostrar estos resultados en la función *calcular()* analizada anteriormente es necesario procesar los datos almacenados en vectores para convertirlos a cadenas. Para ello se emplea el método *join()* para unir los valores del vector en forma de cadena

```

//resultados obtenidos almacenados como cadenas
cadenaAves=vectorAves.join();
cadenaResponsableAves=vectorRAves.join();
cadenaParques=vectorParques.join();
cadenaResponsableParques=vectorRParques.join();
cadenaNombreAeropuerto=vectorNombreAeropuerto.join();
cadenaResponsableAeropuerto=vectorResponsableAeropuerto.join();
cadenaNombrePoblacion=vectorNombrePoblacion.join();

```

De esta forma los datos ya están disponibles para mostrarlos al usuario, solo queda procesar cada mensaje en función del resultado obtenido.

```

//Obtención de la información relativa al id para modificarlo con
nueva información y poder mostrar los resultados mediante un popup
var
resultadosAeropuertos=document.getElementById('resultadoAeropuertos'
);
if(cadenaNombreAeropuerto=="")

```

```

{
resultadosAeropuertos.innerHTML = "La ruta no pasa por ninguna zona
limitada por aeropuertos ";
valida+=1;
}
else{
resultadosAeropuertos.innerHTML = "<strong>La ruta pasa por los
siguientes aeropuertos </strong>" + cadenaNombreAeropuerto +
"<strong> cuyos responsables son </strong>" +
cadenaResponsableAeropuerto;
}

```

Este procedimiento se realiza con todas las capas analizadas, pero en este caso se analiza para los aeropuertos. Como se puede ver en el código anterior, en primer lugar, es necesario obtener la información contenida en el id, en este caso “resultadoAeropuertos” que hace referencia a la sección de aeropuertos del modal mencionado anteriormente. Se comprueba el contenido de la cadena, en caso de que se encuentre vacía significa que la ruta no interseca con ningún aeropuerto por lo que se muestra el mensaje indicándolo. En caso de que no esté vacía supone que la ruta interseca con algún aeropuerto por lo que se inserta un mensaje indicando el nombre del aeropuerto por el que pasa la ruta y el responsable de este.

INSTRUCCIONES.HTML

En este documento se encuentra la estructura y diseño de la página que muestra las instrucciones de uso de la aplicación para facilitar al usuario la toma de contacto con la aplicación y el correcto uso. Esta página solo es accesible desde la página de inicio y las páginas de instrucciones ya que es el momento de que el usuario se familiarizarse y conozca la aplicación antes de utilizarla. Por este motivo la barra de navegación de las páginas de instrucciones y recomendaciones difieren del resto excepto de la página de aplicación, ya que se incluye la navegación mediante un desplegable a las diferentes instrucciones (ver Figura 41).

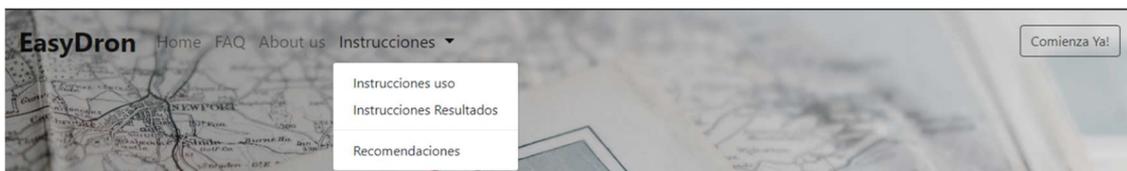


Figura 41: Barra de navegación páginas instrucciones

Como se puede observar en la Figura 42 las instrucciones se indican paso a paso para que resulte más intuitivo al usuario.

Instrucciones de uso EasyDron

1º Dibujar la ruta

Pulsar el botón "dibujar" y trazar sobre el mapa la ruta que se desea dibujar. Con un click de ratón se genera un punto fijo de inicio para realizar la ruta hasta un nuevo punto que puede ser intermedio o final. Con doble click se termina de dibujar la ruta (en este momento el trazado pasa de estar resaltado a reducirse).

2º Editar la ruta

Si la ruta no es la deseada puede editarla pulsando el botón "Editar". Pincha en el punto de la ruta que desea modificar y arrastre a la zona deseada. Cuando este en la posición deseada solo tiene que soltar.

3º Comprobar la ruta

Para calcular si es una ruta válida solo tiene que pulsar el botón "Calcular ruta" (asegurese de que la ruta no se encuentra resaltada, si es así haga doble click en el último punto de la ruta).

3º Espera

Espera que se realicen los cálculos hasta que aparezca un mensaje indicando el fin de estos .

4º Visualización de resultados

Para comprobar los resultados obtenidos solo queda pinchar en "Resultados".

5º Ayuda en la planificación

Puede visualizar la información de los lugares que tienen restricciones de vuelo para ayudar en la planificación de la ruta. Para ello pulse el botón "Capas", aparece una ventana donde seleccionar la información que desea visualizar así como diferentes vistas del mapa. Podrá visualizar Parques Naturales (verde), Zonas de Especial Protección de Aves (Azul), Aeropuertos, helipuertos y aeródromos (Naranja) y Poblaciones (Gris).

Figura 43: Instrucciones de uso EasyDron

INSTRUCCIONESRESULTADOS.HTML

Esta página mantiene las mismas características en la barra de navegación que en el apartado anterior. En cuanto al contenido, trata de dar una explicación de los datos que se muestran al usuario y una ayuda a la hora de aprender a interpretarlos. Para ello se incluye una lista desplegable en la que se explica los resultados de cada apartado, de esta forma el usuario puede acceder rápidamente a la información que desea.

Instrucciones resultados

En el apartado de resultados podrá encontrar información relativa a: Parques Naturales, Aeropuertos, helipuertos y aeródromos, Zonas de Especial Protección de Aves y Poblaciones. Toda esta información tendrá relación con la ruta dibujada anteriormente.

▼ Parques Naturales
▼ Aeropuertos, Helipuertos y Aeródromos
▼ Zonas de Especial Protección de Aves (ZEPAs)
▼ Poblaciones
Se informa de que la ruta no pasa por ninguna zona de limitada al vuelo por pertenecer a Poblaciones, o por el contrario se indica el nombre o nombres de los Poblaciones que sobrevuela. En el caso de tratarse de un dispositivo de 250g o menor puede realizar vuelos bajo ciertas condiciones a baja altura respetando las recomendaciones de la AESA. En caso de duda consulte con el ayuntamiento o responsable de la población. (Recomendaciones AESA)
▼ Resultado final
▼ Resultado visual

Figura 44: Instrucciones para la interpretación de los resultados.

En esta página podemos ver información al respecto de cada una de las zonas en las que puede existir algún tipo de restricción al vuelo, indicando como se muestran los resultados, incluso ofreciendo enlaces para obtener más información sobre dichas zonas. También se indica como interpretar el resultado final y como visualizar el recorrido que interseca con las zonas mencionadas anteriormente para que resulte más fácil la realización de una nueva ruta que evite esta zona.

[RECOMENDACIONES.HTML](#)

Por último, está la página de recomendaciones, en este caso se busca dar al usuario una serie de recomendaciones a la hora de realizar un vuelo con su dron ya que no solo es necesario respetar las limitaciones de ciertas zonas que se indican en la aplicación. También es necesario tener en cuenta otros factores como son la climatología o la ley de protección de datos (LOPD) que afecta a las posibles grabaciones que se puedan realizar con el dron (ver Figura 45).

Preguntas y respuestas frecuentes



Condiciones meteorológicas

En el momento de realizar el vuelo asegúrate de que las condiciones climáticas son las adecuadas: suficiente luz solar para una buena visibilidad sin que pueda producir deslumbramiento, temperatura adecuada que no suponga dificultad en el control del dispositivo y evitar pilotar con fuertes rachas de viento que imposibiliten el control del dron.



Altura de vuelo

Vigilar que la altura de vuelo no sobrepase los 120m ya que supone una infracción del reglamento de vuelo y que no sea demasiado baja para evitar colisionar y causar daños a objetos, obstáculos o instalaciones.



Sobrevolar personas

Evitar sobrevolar personas o multitudes ya en caso de accidente o pérdida de control puede causar daños a estas. Es recomendable poseer un seguro que se haga cargo de los daños causados por el vuelo de drones.



Grabación o toma de imágenes

En el caso de que se realice tomas de fotografías o grabaciones de video mediante el uso del dron durante el vuelo, es necesario tener en cuenta la Ley de Protección de Datos [Ley Orgánica 3/2018](#) en el que se declara el derecho al privacidad de los individuos.



Espacio Aéreo Restringido

A pesar de que el algoritmo de la aplicación tiene en cuenta el espacio reservado para aeropuertos también es necesario tener en cuenta el espacio aéreo restringido. En las proximidades a los aeropuertos es común de que exista este tipo de espacio aéreo restringido por lo que si tienes planeado realizar un vuelo por las proximidades es conveniente contactar con los responsables del aeropuerto.



Precauciones

A pesar de los cálculos realizados por la aplicación es necesario tomar todo tipo de precauciones para evitar cualquier situación de riesgo o peligro.

Figura 45: Recomendaciones para el vuelo

CONCLUSIONES

El uso de tecnologías de Infraestructura de Datos Espaciales o del conjunto de datos que ponen a disposición, como en este caso, permite la creación y el análisis de nuevas aplicaciones empleando para ello geometrías de espacios ya sea en forma de punto líneas o polígonos que pueden ser almacenados y tratados para obtener el objetivo deseado. Esto ha permitido el desarrollo de la aplicación EasyDron que mediante el uso de datos de diferentes zonas ofrecidos por instituciones españolas y datos propios ha sido posible analizar geográficamente la validez de una posible ruta o zona de vuelo de un dron.

Desarrollo cuyo fin es servir de ayuda a cualquier usuario, sobre todo un usuario inexperto y con desconocimiento del ámbito a realizar una ruta cumpliendo la normativa y evitando la mayor cantidad de peligros posibles.

El principal inconveniente encontrado a la hora de realizar el desarrollo de la aplicación ha sido no encontrar un IDE completa proporcionada por las instituciones al que acceder mediante los servicios WFS y WPS por lo que ha sido necesario realizar un tratamiento propio de los datos, lo que da lugar a la necesidad de comprobaciones periódicas para mantener los datos actualizados al usuario.

Por otra parte, con el desarrollo de la aplicación se ha conseguido mostrar al usuario las diferentes zonas con limitaciones o con posibilidad de vuelo mediante el uso de mapas y representación geográfica de las zonas lo que hace más fácil su comprensión. De esta forma junto con los consejos e indicaciones para la realización de un vuelo más seguro se proporciona al usuario de un dron una serie de pautas para un mejor uso de estas aeronaves.

Líneas Futuras

Como hemos visto a lo largo del documento los dispositivos estudiados, los drones, están en auge y se espera que siga aumentando sobre todo en el ámbito industrial, lo que dará lugar a modificaciones de la regulación actual. Estas modificaciones o

ampliaciones pueden dar lugar a actualizaciones de la aplicación incluyendo nuevos parámetros para la realización de los cálculos como puede ser el uso del dron o el modelo de dron utilizado.

Haciendo referencia a casos en estudio y que comienzan a ponerse en marcha como el uso de drones en el sector de la agricultura puede dar lugar a la ampliación o creación de una nueva versión enfocada a este sector, para cubrir nuevas necesidades. Esta ampliación podría cubrir la necesidad de pedir permisos para realizar vuelos en zonas que puedan ser conflictivas debido a la proximidad de un aeropuerto poniendo a disposición de la empresa/usuario de los formularios pertinentes. Por otra parte, también podría servir para llevar un control de las zonas en las que se ha realizado el riego o fumigación mediante el uso de datos recogidos por el dron y registrados en forma de geometría en la base de datos para su posterior análisis.

Otro de los puntos que puede ampliarse está condicionado por el desarrollo de una IDE que ofrezca datos relativos a zonas de espacio aéreo restringido como pueden ser zonas en las que se realizan maniobras para aterrizaje o despegue. Por lo que el uso de estas tecnologías es amplio siendo necesario la recopilación y creación de nueva información que pueda dar lugar al desarrollo de nuevas aplicaciones basadas en este tipo de servicio.

Referencias

- Aguiar, Alberto R. 2020. «Por qué Amazon, UPS o incluso Domino's están invirtiendo en servicios de reparto con drones». Business Insider. 16 de febrero de 2020. <https://www.businessinsider.es/amazon-ups-dominos-invierten-drones-reparto-582105>.
- «BOE-A-2017-15721.pdf». s. f. Accedido 21 de abril de 2020. <https://www.boe.es/boe/dias/2017/12/29/pdfs/BOE-A-2017-15721.pdf>.
- Bullock, Clara. 2020. «Drones to Deliver Defibrillators during Emergencies in Sweden». AirMed&Rescue. 20 de mayo de 2020. <https://www.airmedandrescue.com/latest/news/drones-deliver-defibrillators-during-emergencies-sweden>.
- Córdoba, Diario. 2020. «Agricultura de precisión a través de drones para el olivar cordobés». Diario Córdoba. 13 de febrero de 2020. https://www.diariocordoba.com/noticias/cordobalocal/agricultura-precision-traves-drones-olivar-cordobes_1353853.html.
- «Denunciado por volar en Elche un dron para grabar un evento». 2020. 6 de marzo de 2020. <https://www.diarioinformacion.com/elche/2020/03/06/denunciado-volar-elche-dron-grabar/2242243.html>.
- «DJI: “La industria del dron crece gracias al uso profesional”». 2017. Expansión.com. 13 de noviembre de 2017. <https://www.expansion.com/economia-digital/protagonistas/2017/11/13/5a05a848e5fdeaa6348b456d.html>.
- «Drones para estudiar huracanes -». 2017. Noticias de El tiempo. 10 de septiembre de 2017. <https://noticias.eltiempo.es/drones-para-estudiar-huracanes/>.
- EDCM/EP. 2020. «Buscan al supuesto cocodrilo del Pisuerga con drones, cebos y cámaras termográficas». www.eldigitalcastillalamancha.es. 10 de junio de 2020. <https://www.eldigitalcastillalamancha.es/actualidad/1465507/Buscan-al-supuesto-cocodrilo-del-Pisuerga-con-drones-cebos-y-camaras-termograficas-.html>.
- Europa Press, Diario de. 2020. «Navarra traslada el uso de drones a los proyectos de construcción de obras públicas». 28 de mayo de 2020. <https://www.noticiasdenavarra.com/actualidad/sociedad/2020/05/28/navarra-navarra-traslada-drones-proyectos/1049956.html>.
- Gabriel, Ramón. 2020. «Denuncian el dueño de un dron por elevarlo cerca del aeropuerto de Lleida». elperiodico. 15 de febrero de 2020. <https://www.elperiodico.com/es/sociedad/20200215/denuncian-el-dueno-de-un-dron-por-elevarlo-cerca-del-aeropuerto-de-lleida-7849834>.
- Geoinnova, Asociación. 2017. «Incendios forestales y el uso de drones para combatirlos». *Territorio Geoinnova - SIG y Medio Ambiente* (blog). 30 de mayo de 2017. <https://geoinnova.org/blog-territorio/drones-incendios-forestales/>.
- Gonzalez Gil, Julio. 2018. «Qué es PostgreSQL». OpenWebinars.net. 16 de diciembre de 2018. <https://openwebinars.net/blog/que-es-postgresql/>.
- Granda, Gerardo. 2020. «Los drones que gritan a los chinos por el coronavirus». La Razón. 10 de febrero de 2020. <https://www.larazon.es/internacional/20200210/f6yizl4xefbtxad5uaxfg2c4u.html>.
- Hernández, Natalia. 2019. «Altos vuelos para el sector de los drones en España». *Forbes España* (blog). 11 de septiembre de 2019.

- <https://forbes.es/empresas/52630/altos-vuelos-para-el-sector-de-los-drones-en-espana/>.
- «IDE». s. f. Accedido 14 de junio de 2020. <https://www.ign.es/web/resources/docs/IGNCnig/IDE-Teoria.pdf>.
- Klokian Technologies, Klokian Technologies. 2000. «ETRS89 / UTM Zone 30N - EPSG:25830». 19 de octubre de 2000. <http://epsg.io>.
- . 2015. «WGS 84 / Pseudo-Mercator - Spherical Mercator, Google Maps, OpenStreetMap, Bing, ArcGIS, ESRI - EPSG:3857». 25 de noviembre de 2015. <http://epsg.io>.
- . 2018. «ETRS89 - EPSG:4258». 16 de febrero de 2018. <http://epsg.io>.
- La actualidad, La Actualidad. 2020. «La Policía Local de Lorca controla con drones el correcto cumplimiento por parte de los bañistas de las normas durante este fin de semana en las calas de Punta de Calnegre». *La Actualidad* (blog). 8 de junio de 2020. <http://www.la-actualidad.com/articulo/06082020/la-policia-local-de-lorca-controla-con-drones-el-correcto-cumplimiento-por-parte-de-los-banistas-de-las-normas-durante-este-fin-de-semana-en-las-calas-de-punta-de-calnegre/>.
- «La entrega con drones de Amazon es real: así fue el primer envío a un cliente». 2016. *El Confidencial*. 14 de diciembre de 2016. https://www.elconfidencial.com/tecnologia/2016-12-14/amazon-drones-entregas-envios_1303960/.
- «La Guardia Civil despliega la “policía del aire” para controlar el vuelo ilegal de drones». 2020. *Antena 3 Noticias*. 22 de febrero de 2020. https://www.antena3.com/noticias/sociedad/la-guardia-civil-despliega-su-policia-del-aire-para-controlar-el-vuelo-ilegal-de-drones_202002225e513f120cf2e7d8ac13f32b.html.
- «Las ventas de drones empresariales crecerán un 50% en 2020». 2019. *MuyCanal* (blog). 13 de diciembre de 2019. <https://www.muycanal.com/2019/12/13/ventas-drones-empresariales>.
- López, José Luis Ortega. 2019. «Un dron entrega un órgano para trasplante con éxito por primera vez». *Cinco Días*. 3 de mayo de 2019. https://cincodias.elpais.com/cincodias/2019/05/02/gadgets/1556804475_272548.html.
- Mark Otto, Jacob Thornton, and Bootstrap, Mark Otto, Jacob Thornton, and Bootstrap. s. f. «History». *Bootstrap's History*. Accedido 9 de mayo de 2020. <https://v4-alpha.getbootstrap.com/about/history/>.
- Morales, Aurelio. 2013. «Los 10 formatos GIS vectoriales más populares». *MappingGIS* (blog). 10 de noviembre de 2013. <https://mappinggis.com/2013/11/los-formatos-gis-vectoriales-mas-populares/>.
- Nikola, Tesla. 1898. Method of and apparatus for controlling mechanism of moving vessels or vehicles. United States US613809A, filed 1 de julio de 1898, y issued 8 de noviembre de 1898. <https://patents.google.com/patent/US613809A/en>.
- «OpenLayers — OSGeoLive 13.0 Documentation». s. f. Accedido 11 de mayo de 2020. https://live.osgeo.org/es/overview/openlayers_overview.html.
- Pita, E.V. 2020. «El usuario del dron que invadió el espacio aéreo del aeropuerto de Peinador afronta multas millonarias». *La Voz de Galicia*. 29 de mayo de 2020. https://www.lavozdeg Galicia.es/noticia/sociedad/2020/05/29/usuario-dron-invadio-espacio-aereo-aeropuerto-peinador-afronta-multas-millonarias/0003_202005202005291590749780500.htm.

- Rocasalva, Anna. s. f. «Los incidentes con drones en Catalunya se quintuplican en cuatro años». *elperiodico*. Accedido 21 de abril de 2020. <https://www.elperiodico.com/es/sociedad/20200409/los-incidentes-con-drones-en-catalunya-se-quintuplican-en-cuatro-anos-7922200>.
- Rosa Fernández, De Modo Que Las Estadísticas Pueden Contener Información Más Actual Que La Recogida En El. 2019. «Tema: Industria de drones en España y el mundo». *www.statista.com*. 30 de septiembre de 2019. <https://es.statista.com/temas/3576/industria-de-drones-en-espana-y-el-mundo/>.
- Salvador, Antonio. 2020. «La historia del dron (fantasma) que obligó a cerrar Barajas». *El Independiente*. 7 de febrero de 2020. <https://www.elindependiente.com/politica/2020/02/08/la-historia-del-dron-fantasma-que-obligo-a-cerrar-barajas/>.
- «Top Trends in the Gartner Hype Cycle for Emerging Technologies, 2017». s. f. Accedido 21 de abril de 2020. [//www.gartner.com/smarterwithgartner/top-trends-in-the-gartner-hype-cycle-for-emerging-technologies-2017/](http://www.gartner.com/smarterwithgartner/top-trends-in-the-gartner-hype-cycle-for-emerging-technologies-2017/).
- «Uso de drones en eventos deportivos: el futuro de las transmisiones». 2020. *¿Quieres ser piloto de drones?* (blog). 14 de febrero de 2020. <https://cursodedrones.es/uso-de-drones-en-eventos-deportivos-el-futuro-de-las-transmisiones-televisivas/>.
- Valdeolmillos, Celia. 2020. «La inversión en sistemas robóticos y drones superará los 128.000 millones en 2020». *MuyComputerPRO*. 3 de enero de 2020. <https://www.muycomputerpro.com/2020/01/03/inversion-sistemas-roboticos-drones-2020>.
- Villareal, Antonio. 2019. «Por qué los drones de la DGT no pueden cazarte si vas corriendo más de la cuenta». 28 de agosto de 2019. https://www.elconfidencial.com/tecnologia/2019-08-28/drones-multas-dgt-radares-vacios_2193495/.
- «Web Content Accessibility Guidelines (WCAG) 2.1». s. f. Accedido 9 de mayo de 2020. <https://www.w3.org/TR/2018/REC-WCAG21-20180605/>.
- «Web Feature Service | OGC». s. f. Accedido 22 de abril de 2020. <https://www.ogc.org/standards/wfs>.
- «Web Map Service | OGC». s. f. Accedido 22 de abril de 2020. <https://www.ogc.org/standards/wms>.
- «Web Processing Service | OGC». s. f. Accedido 22 de abril de 2020. <https://www.ogc.org/standards/wps>.