



**Memoria proyecto fin de ciclo**  
**2024-2025**

**AsturDron**

**Llarina Sanz Uría**

**2º DAW**



---

## Índice

---

1. Presentación de AsturDron.....	3
2. La idea.....	3
3. Desarrollo de la idea.....	3
4. A quien va dirigido y objetivos.....	4
5. Análisis de costes.....	5
6. Plan de financiación.....	6
7. Plan de RRHH.....	6
8. Plan de prevención de riesgos laborales.....	7
9. Requisitos.....	8
10. Diseño.....	13
11. Implementación.....	14
12. API AsturDron.....	15
13. Front-end.....	20
14. Conclusión.....	29
15. Bibliografía.....	30



---

## *1. Presentación de AsturDron.*

---

AsturDron es una aplicación creada para ayudar a los usuarios de drones a saber cuándo pueden volar o cuando no dependiendo del tiempo que haga, tener a mano las noticias más recientes del mundo dron e información relevante para volar, además tiene un apartado para entretener a los usuarios recreando fotos y videos con el fin de entrar en el ranking de usuarios gestionado por el administrador, y por último habrá un apartado en el que los usuarios podrán publicar anuncios de sus drones para que otros usuarios los puedan ver y comentarle para ponerse de acuerdo en el precio y el envío (que correrá por parte del usuario).

---

## *2. La idea.*

---

Esta idea surgió de una afición propia, basándome en las funcionalidades que encuentro necesarias al momento del vuelo o que me parecen interesantes y entretenidas. Me pareció una buena idea basar mi proyecto en los drones ya que cada vez hay más usuarios de éstos en Asturias, ya no son solo utilizados por profesionales, sino también por particulares para su entretenimiento y esto es algo en parte peligroso, ya que muchos de estos usuarios no saben las reglas que deben cumplir. Así nace AsturDron para mejorar la seguridad de los ciudadanos y brindarles una experiencia de vuelo agradable y sin riesgos.

---

## *3. Desarrollo de la idea.*

---

Se trata de crear una aplicación web que permita a los usuarios de ésta acceder a las noticias e información más reciente, podrán consultar también el tiempo para los próximos cuatro días mostrando un mensaje de peligro si el tiempo para ese día es incompatible con el vuelo, también podrán acceder a una biblioteca de videos e imágenes, pudiendo publicar cada usuario sus fotos y videos para que el administrador los puntúe y puedan entrar en el ranking de usuarios y por último también podrán acceder a un apartado de anuncios para publicar los drones que ya no usen y que el resto de usuarios se lo puedan comprar (este apartado es meramente informativo, no se realizará la acción de compraventa por parte de la app sino del usuario por su cuenta).



En caso de estar registrado, el usuario podrá acceder a todas las funcionalidades de la web, mientras que, si no lo está, no podrá acceder a las imágenes y videos ni tampoco publicarlos en el apartado de ocio, pero si podrá ver las noticias e información y el tiempo en Asturias para los próximos 4 días.

Por otra parte, el administrador, además de poder acceder a todas las funcionalidades, como es lógico, podrá eliminar usuarios si no cumplieran las normas de comunidad, modificar o publicar nuevas noticias y, por último, publicar y puntuar fotos y videos.

---

#### *4. A quién va dirigido y objetivos.*

---

AsturDron va dirigido, como ya he dicho previamente, a todos los asturianos que posean un dron, sea para el fin que sea (seguridad, ocio, etc...), con el objetivo de que puedan tener a mano toda la información necesaria para el vuelo y además divertirse replicando fotos y videos del apartado de ocio para que se les puntúe y entrar al ranking.

Algunos de los objetivos más específicos son:

1. **Desarrollo Tecnológico:** Desarrollar eficaz y eficientemente la página web, asegurando que funcione correctamente y para todo tipo de dispositivos gracias a su diseño responsivo.
2. **Mantener actualizada la página:** El administrador (yo) deberá mantener actualizada la página para que no acabe en desuso por quedar obsoleta, actualizando los datos de la página mínimo una vez al mes.
3. **Interfaz atractiva:** Mantener la interfaz de la página atractiva y llamativa, para que la experiencia sea memorable para cualquier usuario y sigan utilizando AsturDron.
4. **Noticias, información y datos meteorológicos fiables:** El administrador deberá publicar solo noticias e información fiable obtenida de la Agencia Estatal de Seguridad Aérea (AESA, 2025) y actualizar los datos meteorológicos de acuerdo con la Agencia Estatal de Meteorología (AEMET, 2025).



---

### 5. *Análisis de costes.*

---

Para la realización de este proyecto pongamos que el único trabajador implicado será yo y trabajaré desde casa 6 horas diarias, 5 días a la semana (excepto festivos) durante los meses de abril mayo y junio de 2025. La previsión de costes de AsturDron grosso modo será la siguiente (8754,23):

- **Gastos de alquiler/hipoteca:** 250€ mensuales.
- **Gastos de luz, gas y agua:** 250€ mensuales.
- **Gastos de internet:** 80€ mensuales.
- **Licencias:** 22,95€ mensuales (Microsoft office 2024 profesional). 187,67€ anuales (Adobe acrobat Standart).
- **Equipo (ordenador HP all-in-one):** 1200€.
- **Sueldo del trabajador:** 5490€
- **Precio del servidor:** 19,57€ mensuales (Deno PRO).

Después de este periodo de 3 meses, realizaré una labor de mantenimiento en la que cada 2 meses trabajaré una semana en resolver errores encontrados en este periodo de tiempo por los usuarios, mejorar y actualizar AsturDron. Esto quiere decir que el precio calculado no será el precio final, sino que seguirá habiendo gastos una vez finalizada la creación de la página web.



---

## 6. Plan de financiación

---

Para financiar este proyecto, inicialmente aportaré la cantidad de 5000€ y pediré un crédito de otros 5000€, la suma de estas dos cifras resulta un poco más de lo previsto en el análisis de costes para afrontar todas las deudas que pueda ocasionar AsturDron hasta que empiece a generar por si solo.

Teniendo en cuenta, que la suscripción a la aplicación costará 7,99€ mensuales si obtuviera una cantidad de 150 usuarios registrados el primer mes, obtendría 1.198,5€ que al año serían 14,382€ (suponiendo que tengamos 150 usuarios durante todo el año, lo cual podría variar a lo largo del año) obteniendo una ganancia de 4.382€ ya que con los otros 10000€ recuperaría mi gasto inicial y pagaría el crédito (sin tener en cuenta los intereses).

Con la ganancia obtenida este primer año, podré empezar a promocionar mi aplicación con merchandising, publicidad pagada, anuncios y colaboraciones con grandes marcas, llegando así a más públicos, pudiendo ampliar la localización de mi aplicación a toda España, obtener más usuarios y, por consiguiente, mayores ganancias.

---

## 7. Plan de RRHH

---

Para el plan de recursos humanos tengamos en cuenta que el equipo está formado únicamente por un desarrollador full stack (yo), cuyas responsabilidades son:

- Análisis de requisitos
- Diseño de arquitectura
- Desarrollo frontend y backend
- Pruebas y control de calidad
- Mantenimiento
- Documentación técnica y de usuario



La carga de trabajo se organizará en 30h semanales distribuidas en:

- 20% análisis y diseño
- 50% desarrollo
- 15% pruebas y QA
- 15% documentación

Las herramientas de trabajo serán:

- Control de versiones: Git + GitHub.
- Entorno de desarrollo: VS Code, IntelliJ, MySQL.
- Documentación: Word.

Habrà una revisión semanal que constará de una evaluación de avances y planificación de la semana siguiente.

---

### *8. Plan de prevención de riesgos laborales*

---

Ya que el trabajo lo realizaré de forma remota, los riesgos laborales descienden de manera significativa.

- **Riesgos identificados:**
  - Sobrecarga de trabajo
  - Falta de motivación o aislamiento
  - Problemas de salud (postura, vista, etc.)
- **Medidas preventivas:**
  - Pausas activas para prevenir el agotamiento
  - Horarios definidos
  - Contacto regular con otros profesionales
  - Proveer una silla y mesa ergonómicas, una buena luz para que no se canse la vista, unas gafas de filtro de luz azul para proteger los ojos y realizar un curso sobre posturas ergonómicas para prevenir dolores de espalda.



---

## 9. Requisitos

---

La especificación de requisitos es un paso importante en el desarrollo de una aplicación web, ya que define las necesidades y expectativas de los usuarios. Es por eso por lo que vamos a dedicar una sección completa a detallar los requisitos necesarios para el correcto funcionamiento de la aplicación, asegurando que se cumplan los objetivos del proyecto y se brinde una experiencia satisfactoria al usuario.

---

### 9.1. Requisitos específicos

---

Para detallar los requisitos hay que seguir una pauta de requerimientos, estos se conforman por: Requerimientos funcionales, requerimientos de interfaces externas, requerimientos de rendimiento, obligaciones del diseño, y atributos.

A continuación, me dispongo a detallarlos.

---

#### 9.1.1. Requerimientos Funcionales

---



##### **Gestión de usuarios:**

- Registro de nuevos usuarios.
- Inicio de sesión para acceder a las funcionalidades de la página como usuario o administrador.
- Eliminar usuarios si no cumplen las normas por parte del administrador.



##### **Gestión de noticias:**

- Creación y edición de noticias llevadas a cabo por el administrador.
- Visualización de noticias por el usuario tanto común como sin registrar.



##### **Gestión de Imágenes:**

- Visualizar, publicar, eliminar y puntuar imágenes por parte del administrador.
- Visualizar y publicar imágenes desde el usuario.





#### **Gestión de Videos:**

- Visualizar, publicar, eliminar y puntuar videos por parte del administrador.
- Visualizar y publicar videos desde el usuario.

#### **Gestión de ranking de usuarios:**

- Visualizar ranking de usuarios completo por todos los usuarios (comunes, sin registrar y administrador).
- Visualizar Top 3 usuarios completo por todos los usuarios (comunes, sin registrar y administrador).

#### **Gestión de anuncios de compraventa:**

- Visualizar, publicar y comentar anuncios por parte del usuario.
- Eliminar anuncio por parte del usuario creador del mismo.
- Visualizar, publicar y eliminar anuncios por parte del administrador.

#### **Gestión de datos meteorológicos:**

- Visualizar datos meteorológicos y mensaje de advertencia si es que hay por parte de usuarios comunes, sin registrar y administrador

---

### *9.1.2. Requerimientos de interfaces externas*

---

La interfaz de usuario debe ser adaptable a todo tipo de usuarios, lo que se conoce como interfaz sencilla e intuitiva, además debe ser responsiva para poder ser accesible desde cualquier dispositivo.



---

#### 9.1.2.1. *Interfaces de usuario*

---

La interfaz de usuario abarca la parte visible y accesible de la app, como ya hemos dicho, es importante que la interfaz sea amigable para dar accesibilidad. Estos son sus requisitos específicos:

- **Registro de usuarios:** Constará de un formulario que recoge los datos del usuario y los almacena en la BBDD siempre y cuando cumpla con las validaciones (dirección email válida y que no esté activa en la página y nombre de usuario que esté inactivo también), creando un nuevo usuario.
- **Inicio de sesión:** También habrá un formulario que recoge los datos introducidos y comprueba que ese usuario esté dado de alta para poder iniciar sesión.

---

#### 9.1.2.2. *Interfaces hardware*

---

Para este proyecto es importante tener en cuenta todos los dispositivos desde los cuales se pueda acceder a esta app, por tanto, esta debe ser compatible con tabletas, móviles y ordenadores con diferentes tamaños de pantallas, esto lo conseguiremos con un diseño responsivo de la página, garantizando una experiencia óptima y agradable. Solo necesitaremos como hardware un dispositivo moderno con acceso a internet para poder acceder a AsturDron.

---

#### 9.1.2.3. *Interfaces software*

---

En cuanto a software, solo necesitaremos que nuestro dispositivo tenga un navegador moderno capaz de ejecutar nuestra página.



---

### 9.1.3. *Requerimientos de rendimiento*

---

- **Tiempo de Respuesta:** La aplicación debe cargar la página principal en menos de 3 segundos.
- **Escalabilidad:** El sistema debe poder llegar a manejar hasta 500 usuarios activos simultáneamente sin degradación del rendimiento. Este número de usuarios es importante para que la aplicación llegue a ser totalmente rentable.
- **Disponibilidad:** La aplicación debe estar operativa y accesible para los usuarios el 99.9% del tiempo durante un año.

---

### 9.1.4. *Obligaciones del diseño*

---

La aplicación debe cumplir con las normas de accesibilidad web (WCAG, 2025).

---

#### 9.1.4.1. *Limitaciones hardware*

---

Realmente no existe más limitación hardware que la tenencia de un dispositivo con acceso a internet y un navegador compatible con nuestra aplicación.

---

### 9.1.5. *Atributos*

---

#### 9.1.5.1. *Facilidades de mantenimiento*

---

Estas facilidades se centran en partir de un código limpio, ordenado, documentado y bien estructurado para que sea comprensible. Como control de versiones utilizamos Git + GitHub para gestionar el desarrollo de AsturDron.



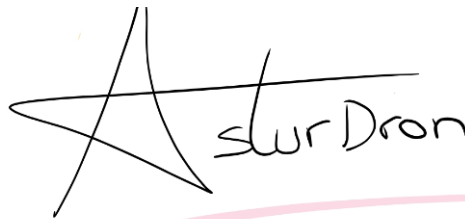
### 9.1.5.2. Portabilidad

La app debe ser portable entre servidores y fácilmente desplegable en distintas plataformas.

### 9.1.5.3. Otros requerimientos

- **Usabilidad:** La interfaz debe ser intuitiva y fácil de usar además de atractiva para favorecer la experiencia del usuario.
- **Fiabilidad:** El sistema debe ser robusto y capaz de solucionar los problemas rápidamente

## 9.2. Diagrama Entidad Relación



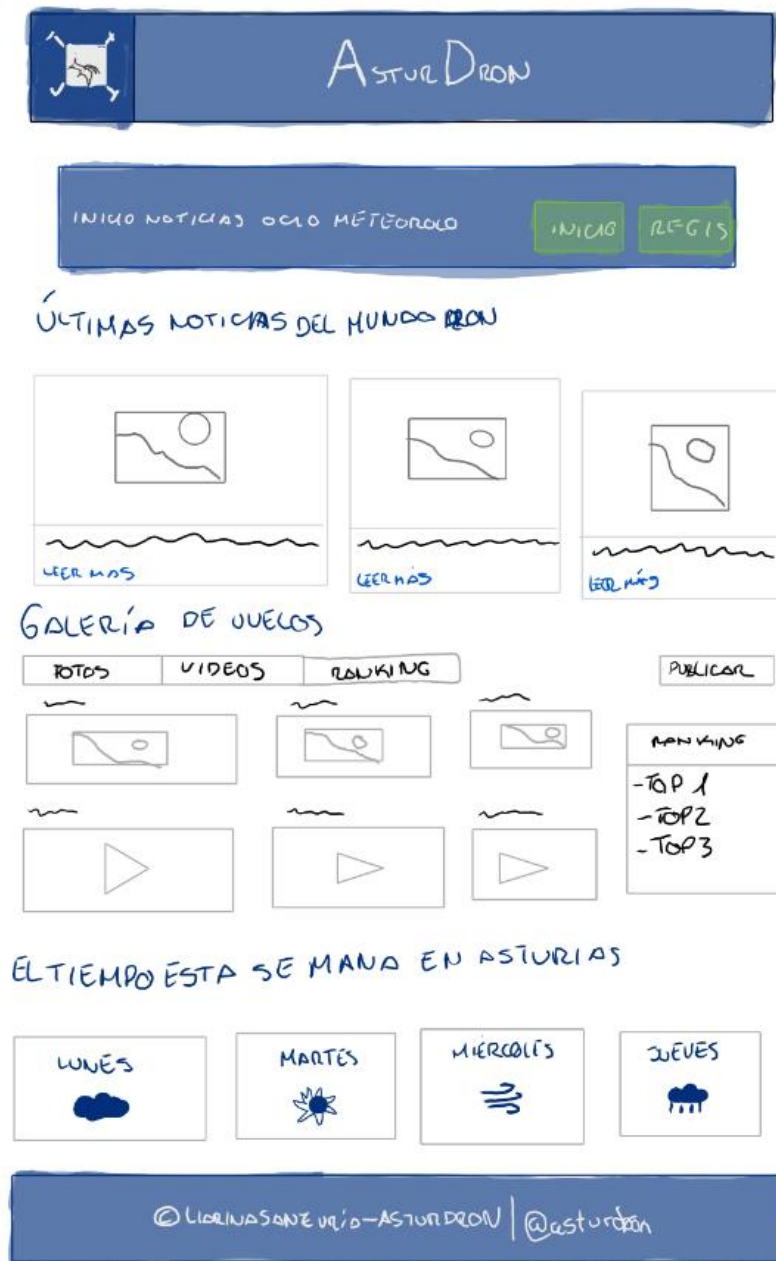
BBDD

- USERS : ID, USERNAME, PASSWORD, LICENSE (01, 02, 03, any), USER\_EMAIL, SCORE ✓
- NOTICES: ID, MINISTURE, TITULAR, NOTICE, LICENSE (YK), DATE YEAR ✓
- IMAGES: ID, USERNAME(YK), IMAGE
- VIDEOS: ID, MINISTURE, VIDEO, USERNAME(YK)
- WEATHER: ID, WEEK\_DAY, DAY, WEATHER ✓



## 10. Diseño

El diseño principal de AsturDron será algo como esto (este es el mockup inicial, luego podrá haber cambios a lo largo del desarrollo de la aplicación):





---

## 11. Implementación

---

En esta fase hay que decidir las tecnologías a utilizar durante el desarrollo de la aplicación:

### 1. A nivel front-end:

- Angular: Angular va a servir para enlazar el cliente con el servidor mediante peticiones.
- HTML5: Con HTML crearé la base de la interfaz.
- CSS: Esta otra tecnología enlaza con HTML para darle estilo a la interfaz y hacerla llamativa visualmente para el usuario.

### 2. A nivel back-end:

- SpringBoot:
  - ◆ API Rest: Esta parte es esencial ya que llevará a cabo la comunicación con el cliente.
  - ◆ Spring Security: proporciona autenticación, autorización y protección contra ataques comunes hacia las contraseñas de los usuarios.
  - ◆ Json Web Token: Trabaja junto a Spring Security para garantizar la transmisión segura de información entre las partes implicadas.
  - ◆ JPA: para realizar operaciones CRUD (crear, leer, actualizar, eliminar) en el servidor.

### 3. BBDD:

- MySQL: Para almacenar toda la información necesaria (usuarios, contraseñas e información acerca de la aplicación) en bases de datos relacionales.



---

## 12. API AsturDron – Estructura y Funcionamiento

---

Este apartado describe la estructura y funcionamiento de la API desarrollada en SpringBoot.

La API de AsturDron está organizada en varias capas, mediante una arquitectura conocida como “Modelo, vista, controlador”, estas capas son:

### 1. Controlador/Controller

En esta capa es donde recibimos los datos de entrada desde el front-end y su función es procesar las peticiones entrantes, realizar validaciones iniciales y delegar la lógica de negocio a la capa de servicio.

Es decir, el controlador es el que comunica el cliente con la lógica del servicio (que estará almacenada en la capa de servicio explicada a continuación).

### 2. Servicio/Service

Esta capa contiene la lógica de la API, gestiona todas las operaciones que queramos realizar e involucra los repositorios u otros servicios para realizar las operaciones solicitadas por el controlador. Estos datos serán procesados antes de interactuar con la BBDD.

En resumen, aquí se realiza la lógica de la API y se encarga además de comunicar el controlador con el repositorio (explicado a continuación).

### 3. Repositorio/Repository

Esta es la capa encargada de interactuar con la BBDD y hacerle las consultas necesarias para el servicio, permitiendo realizar consultas personalizadas más complejas que las que ya incluye JPA o CRUD.

### 4. Modelo/Model

En esta capa se definirán las entidades que representan las tablas de la BBDD, estas especifican las relaciones entre tablas y mapeos.

### 5. DTO

Esta capa ayuda al servicio a filtrar los datos que deseamos obtener del modelo, es decir, si tengo 5 campos en el modelo, pero solo necesito 3 de ellos en la salida, utilizo esta capa como intermediaria.



---

## 12.1. Gestión de usuarios

---

### Endpoints y funcionalidades

#### 1. Registro de usuarios:

- **Método:** POST
- **URL:** </ApiRestBackEnd/auth/register>
- **Descripción:** Permite a un usuario registrarse proporcionándole los datos solicitados por el formulario.

#### 2. Autenticación de usuarios (login):

- **Método:** POST
- **URL:** </ApiRestBackEnd/auth/login>
- **Descripción:** Permite a un usuario iniciar sesión , autenticando si este existe o no.

#### 3. Eliminar usuarios:

- **Método:** DELETE
- **URL:** </ApiRestBackEnd/User/username/{username}>
- **Descripción:** Permite al administrador eliminar por su nombre de usuario a un usuario si este incumple la normativa.

#### 4. Obtener usuarios para panel de administración:

- **Método:** GET
- **URL:** </ApiRestBackEnd/User>
- **Descripción:** Permite al administrador obtener todos los usuarios con todos sus datos para poder gestionarlos.

#### 5. Obtener usuarios para panel de administración:

- **Método:** GET
- **URL:** </ApiRestBackEnd/User>
- **Descripción:** Permite al administrador obtener todos los usuarios con todos sus datos para poder gestionarlos.





#### 6. Obtener ranking de usuarios:

- **Método:** GET
- **URL:** </ApiRestBackEnd/User/ranking>
- **Descripción:** Permite al administrador obtener los datos necesarios de todos los usuarios con su puntuación ordenados de mayor a menor.

#### 7. Obtener Top 3 del ranking de usuarios:

- **Método:** GET
- **URL:** </ApiRestBackEnd/User/ranking/top3>
- **Descripción:** Permite al administrador obtener los 3 primeros usuarios del ranking con los datos necesarios.

---

### 12.2. Gestión de noticias

---

#### 8. Obtener noticias por licencia:

- **Método:** GET
- **URL:** </ApiRestBackEnd/Notice/license/{license}>
- **Descripción:** Obtiene las noticias por la licencia introducida.

#### 9. Obtener todas las noticias:

- **Método:** GET
- **URL:** </ApiRestBackEnd/Notice>
- **Descripción:** Obtiene todas las noticias.

#### 10. Crear noticias:

- **Método:** POST
- **URL:** </ApiRestBackEnd/Notice>
- **Descripción:** Permite al administrador crear nuevas noticias.

#### 11. Actualizar noticias:

- **Método:** UPDATE
- **URL:** </ApiRestBackEnd/Notice/{id}>
- **Descripción:** Permite al administrador actualizar los datos de las noticias existentes por id.



## 12. Eliminar noticias:

- **Método:** DELETE
- **URL:** /ApiRestBackEnd/Notice/{id}
- **Descripción:** Permite al administrador eliminar noticias existentes y obsoletas.

---

### 12.3. Gestión de imágenes

---

## 13. Obtener imágenes:

- **Método:** GET
- **URL:** /ApiRestBackEnd/images
- **Descripción:** Obtiene las imágenes publicadas por todos los usuarios y por el administrador.

## 14. Publicar imágenes desde local:

- **Método:** POST
- **URL:** /ApiRestBackEnd/images/upload-file/{username}
- **Descripción:** Permite al usuario o al administrador publicar una imagen desde local.

## 15. Publicar imágenes desde url:

- **Método:** POST
- **URL:** /ApiRestBackEnd/images/upload/{username}
- **Descripción:** Permite al administrador publicar una imagen desde una url.

## 16. Eliminar imágenes:

- **Método:** DELETE
- **URL:** /ApiRestBackEnd/images/{id}
- **Descripción:** Permite al administrador eliminar una imagen por su id.

## 17. Puntuar imágenes:

- **Método:** PUT
- **URL:** /ApiRestBackEnd/images/{id}/score
- **Descripción:** Permite al administrador puntuar una imagen.



---

## 12.4. Gestión de Videos

---

### 18. Obtener videos:

- **Método:** GET
- **URL:** [/ApiRestBackEnd/images](#)
- **Descripción:** Obtiene las imágenes publicadas por todos los usuarios y por el administrador.

### 19. Publicar videos desde local:

- **Método:** POST
- **URL:** [/ApiRestBackEnd/videos/upload-file/{username}](#)
- **Descripción:** Permite al usuario o al administrador publicar un video desde local.

### 20. Publicar videos desde url:

- **Método:** POST
- **URL:** [/ApiRestBackEnd/videos/upload/{username}](#)
- **Descripción:** Permite al administrador publicar un video desde una url.

### 21. Eliminar videos:

- **Método:** DELETE
- **URL:** [/ApiRestBackEnd/videos/{id}](#)
- **Descripción:** Permite al administrador eliminar un video por su id.

### 22. Puntuar videos:

- **Método:** PUT
- **URL:** [/ApiRestBackEnd/videos/{id}/score](#)
- **Descripción:** Permite al administrador puntuar un video.



## 12.5. Gestión del Tiempo

### 23. Obtener datos meteorológicos:

- **Método:** GET
- **URL:** /ApiRestBackEnd/Weather
- **Descripción:** Obtiene los datos meteorológicos de los 4 próximos días contando desde el día actual.

## 13. Front-End y diseño de la interfaz

Para la implementación del front-end he decidido utilizar angular ya que es el framework en el que mejor me desenvuelvo. Para llevarlo a cabo he tenido que crear varios componentes:

### 1. Header coponent:

Componente que contiene la vista del encabezado de la página



### 2. Auth component:

- **Login:** recoge todas las funcionalidades, la vista y el estilo del botón y formulario de inicio de sesión.



### Iniciar Sesión

Usuario:

Contraseña:

Iniciar Sesión



- **Register:** recoge todas las funcionalidades, la vista y el estilo del botón y formulario de registro de usuarios.



### Registrarse

Usuario:

Email:

Contraseña:

Licencia:

Sin licencia

▼

Registrarse

### 3. Home component:

Este componente almacena todo el contenido (vista) principal de la página, que está almacenado aun así en otros componentes.



#### Últimas noticias del mundo dron



Mostrando 1 - 2 de 2 noticias





### Galería de vuelos

Fotos

Videos

Ranking

Publicar Foto

Para ver las imágenes con claridad, debes [iniciar sesión](#) o [registrarte](#)

Mostrando 1 - 2 de 7 imágenes

@admin



@admin



← Anterior

1

2

3

4

Siguiente →

Página 1 de 4

### El tiempo esta semana en asturias

**Viernes**

**6**



Sunny

**Sábado**

**7**



Cloudy

**Domingo**

**8**



Rainy

**Lunes**

**9**



Sunny



#### 4. Upload-modal component:

Este es el que lleva a cabo la subida de archivos (imágenes).

**Subir Foto**

⚠ Debes iniciar sesión para publicar contenido

Nombre de usuario:  
Introduce tu nombre de usuario

Archivo de imagen:  
Seleccionar archivo Ningún archivo seleccionado

Debes iniciar sesión

---

**Subir Foto**

Nombre de usuario:  
tino

Archivo de imagen:  
Seleccionar archivo dron2.jpg

dron2.jpg (3.73 KB)

Previsualización:  


Subir Foto



## 5. Video-modal component:

Este es el que lleva a cabo la subida de archivos (videos).

**Subir Video**

⚠ Debes iniciar sesión para publicar contenido

Nombre de usuario:  
Introduce tu nombre de usuario

Archivo de video:  
Seleccionar archivo Ningún archivo seleccionado

Archivo de miniatura (imagen):  
Seleccionar archivo Ningún archivo seleccionado

Debes iniciar sesión

---

**Subir Video**

Nombre de usuario:  
llari

Archivo de video:  
Seleccionar archivo 13482389\_2732\_1440\_30fps.mp4  
13482389\_2732\_1440\_30fps.mp4 (12.01 MB)

Archivo de miniatura (imagen):  
Seleccionar archivo dron2.jpg  
dron2.jpg (3.73 KB)

Subir Video

## 6. Guards component:

Obtiene el nombre del usuario registrado para publicar una foto o video.

**Subir Foto**

Nombre de usuario:  
llari

Archivo de imagen:  
Seleccionar archivo Ningún archivo seleccionado

Completa los campos






## 7. Noticias componenet:

Este es el componente que solicita las noticias al backend y muestra su vista y estilos.

### Últimas noticias del mundo dron


**Filtrar por licencia:** A1 A2 A3 Todas

Mostrando 1 - 2 de 3 noticias



Noticia prueba 1

[Leer más](#)



Noticia prueba 2

[Leer más](#)

← Anterior 1 2 Siguiente →

Página 1 de 2


## 8. Meteorología component:


Este es el componente que solicita los datos meteorológicos al backend y muestra su vista y estilos.


### El tiempo esta semana en asturias

**⚠️ PROHIBIDO VOLAR HOY: Condiciones meteorológicas adversas**

**Viernes**  
**6**  
  
Windy





**Sábado**  
**7**  
  
Pendiente

**Domingo**  
**8**  
  
Pendiente

**Lunes**  
**9**  
  
Pendiente



## El tiempo esta semana en asturias

<b>Viernes</b> <b>6</b>  Sunny	<b>Sábado</b> <b>7</b>  Cloudy	<b>Domingo</b> <b>8</b>  Rainy	<b>Lunes</b> <b>9</b>  Sunny
--	--	--	--

## 9. Ocio component:


Este gestiona y muestra la vista y estilos de las fotos, videos y ranking de usuarios.


### Galería de vuelos

[Fotos](#) [Videos](#) [Ranking](#) [Publicar Foto](#)

Para ver las imágenes con claridad, debes [iniciar sesión](#) o [registrarte](#)

Mostrando 1 - 2 de 5 imágenes

@admin  


@admin  


[< Anterior](#) [1](#) [2](#) [3](#) [Siguiente >](#)

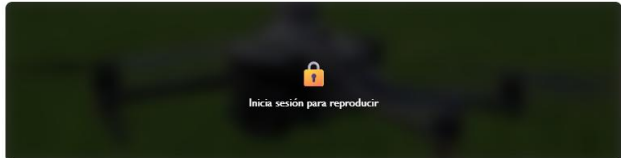
Página 1 de 3

### Galería de vuelos

[Fotos](#) [Videos](#) [Ranking](#) [Publicar Video](#)

Para reproducir videos, debes [iniciar sesión](#) o [registrarte](#)

Mostrando 1 - 1 de 1 videos

@admin  




### Galería de vuelos

Fotos Videos Ranking Publicar Foto

Mostrando 1 - 2 de 5 imágenes

@admin



@admin



← Anterior

1

2

3

Siguiente →

Página 1 de 3

### Galería de vuelos

Fotos Videos Ranking Publicar Vídeo

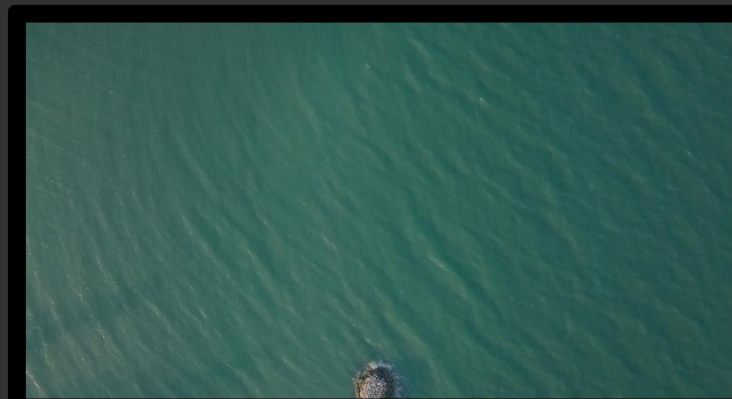
Mostrando 1 - 1 de 1 videos

@admin



### Galería de vuelos

Fotos Videos Ranking Publicar Vídeo





### Galería de vuelos

Fotos

Videos

Ranking

Ver Ranking Completo

### Top 3 Jugadores

Clasificación de usuarios por puntuación

Posición	Usuario	Puntuación
1	llari	105
2	tino	13
3	denis	0

### Galería de vuelos

Fotos

Videos

Ranking

Ver Top 3

### Ranking Completo

Clasificación de usuarios por puntuación

Posición	Usuario	Puntuación
1	llari	105
2	tino	13
3	denis	0
4	pepe	0

## 10. Footer component:

Muestra la vista y los estilos del pie de página de la app.



---

### 14. Conclusión

---

AsturDron es una app de interfaz sencilla e intuitiva y muy fácil de manejar, por lo que es accesible para todo tipo de usuarios, ésta contiene información necesaria (datos meteorológicos, noticias y normativa) para pilotos de drones con o sin licencia de vuelo, también tiene un apartado de ocio, donde podemos encontrar imágenes y videos que los usuarios podrán replicar con sus dispositivos, además hay ejemplos de lugares bonitos para sobrevolar y fotografiar con un estos dispositivos (disponibilidad únicamente para Asturias) y por último un apartado de compra-venta de drones entre usuarios. En cuanto al desarrollo de la app, será sencillo ya que se trata de un simple manual en el que el administrador solo necesita investigar en la Agencia Estatal Meteorológica (AEMET, 2025) y la Agencia Estatal de Seguridad Aérea (AESA, 2025) para obtener los datos necesarios para su desarrollo, la única complejidad que acarrea será la notificación de condiciones meteorológicas no aptas para el vuelo y la conexión entre el usuario y el administrador para la valoración de los videos e imágenes, además del contacto entre usuarios para comprar o vender sus dispositivos.



---

### *15. Bibliografía*

---

AEMET: [Agencia Estatal de Meteorología - AEMET. Gobierno de España](#)

AESA: [AESA-Agencia Estatal de Seguridad Aérea - MTMS](#)