

# Analysis report D02

**Group:** C1.060

**Repository:** <https://github.com/DP2-C1-060/Acme-ANS-D02>

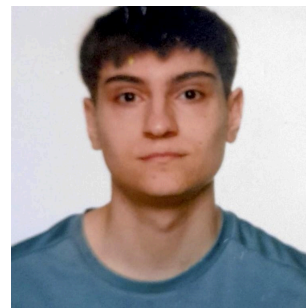
Student #1

**ID:** 2062\*\*\*\*T

**UVUS:** javmunrom

**Name:** Muñoz Romero, Javier

**email:** javmunrom@alum.us.es



**Roles:** Analista, Desarrollador

**Date:** Universidad de sevilla Marzo 13, 2025

<b>1. Resumen Ejecutivo</b>	<b>2</b>
<b>2. Tabla de Revisiones</b>	<b>2</b>
<b>3. Introducción</b>	<b>2</b>
Task 03: Manage airline managers data	2
Descripción:	2
Análisis:	3
Task 04: Store and handle flight data	3
Descripción:	3
Análisis:	3
Descripción:	4
Análisis:	4
Task 06: Generate sample test data	5
Descripción:	5
Análisis:	5
Task 07: Provide GitHub planning link	6
Descripción:	6
Análisis:	6
Task 15: Implement manager dashboards	6

Descripción:	6
Análisis:	6
Task 16: Design UML domain model	7
Descripción:	7
Análisis:	7
Task 26: Integrate weather data service	8
Descripción:	8
Análisis:	8
Task 27: Produce analysis report	8
Descripción:	8
Análisis:	8
Task 28: Generate planning & progress report	8
Descripción:	8
Análisis:	8
<b>5. Conclusiones</b>	<b>9</b>

## 1. Resumen Ejecutivo

Este informe recoge de manera detallada el análisis realizado sobre los requisitos del proyecto, incluyendo las modificaciones necesarias, las decisiones adoptadas para cumplir con los objetivos establecidos y los enlaces a las validaciones de los docentes. Se describe cómo se han implementado las funcionalidades y se integra la verificación de datos a lo largo del desarrollo, asegurando un producto final robusto y coherente.

## 2. Tabla de Revisiones

Número de revisión	Fecha	Descripción de la revisión
1.0	13/03/2025	Versión inicial del documento

## 3. Introducción

El presente documento detalla el análisis llevado a cabo sobre aquellos requisitos del proyecto que presentaban ambigüedades o necesitaban ajustes. En este análisis se exponen las modificaciones implementadas, las decisiones adoptadas y las validaciones confirmadas por el profesorado.

El objetivo de este informe es demostrar, de manera clara y concisa, cómo cada uno de los requisitos se ha abordado en el desarrollo de la aplicación. Se presta especial atención a la

integración de funcionalidades críticas, tales como la gestión de datos de managers, vuelos y etapas, el seguimiento de condiciones meteorológicas y la generación de dashboards para la toma de decisiones estratégicas.

## 4. Análisis de Requisitos

### Task 03: Manage airline managers data

#### Descripción:

Airline managers are the people responsible for managing flights. The system must handle the following information about managers: an identifier number (unique, pattern "[A-Z]{2-3}\d{6}\$", where the first two or three letters correspond to their initials), years of experience in the airline, his or her date of birth and an optional link to a picture that must be stored somewhere else.

#### Análisis:

En esta tarea se ha completado la implementación de la gestión de datos de los responsables de aerolíneas, cumpliendo con los requisitos establecidos de forma integral. Se han aplicado las siguientes validaciones y características:

- Validación del identificador: Se asegura que el identificador siga un patrón específico (dos o tres letras seguidas de seis dígitos) y, además, se verifica que los dos primeros caracteres coincidan con las iniciales del nombre y apellido del responsable, garantizando coherencia en la identificación.
- Control de unicidad: Se implementa la comprobación para que cada identificador sea único en el sistema, evitando la existencia de duplicados y manteniendo la integridad de los datos.
- Verificación de la edad y años de experiencia: Se comprueba que la fecha de nacimiento sea anterior al momento actual, determinando la edad del responsable. Se exige que tenga, como mínimo, 16 años y se valida que los años de experiencia no superen el máximo permitido, calculado como la diferencia entre la edad y los 16 años.
- Validaciones adicionales: Se incorporan restricciones para datos opcionales, como el enlace a una imagen, asegurando que se respete el formato correcto.

En resumen, se ha logrado implementar de forma completa el requisito de gestión de datos de los managers, integrando validaciones tanto a nivel de formato como de consistencia y unicidad, lo que permite contar con información fiable y estructurada en el sistema.

### Task 04: Store and handle flight data

#### Descripción:

A flight is a scheduled journey made by airlines to transport passengers between two locations. The system must store the following data about them: a tag that highlights some

feature of the flight such as "the fastest", "the cheapest" (up to 50 characters), an indication on whether it requires self-transfer or not, a cost, an optional description (up to 255 characters). It also stores information that comes from its legs, namely: a scheduled departure and a scheduled arrival that depends on the first scheduled departure moment of the first leg and the scheduled arrival moment of the last leg, the origin and destination cities that come from the city of the airports to which the first and last leg refer to, and finally, the number of layovers.

#### Análisis:

En esta tarea se ha completado la implementación del almacenamiento y gestión de los datos de los vuelos, cumpliendo con lo establecido en el requisito. Se ha definido que cada vuelo debe incluir:

- Datos básicos: Se almacena un tag descriptivo (limitado a 50 caracteres) que destaca alguna característica del vuelo, un indicador booleano de self-transfer, el coste (con restricciones en su rango) y una descripción opcional (hasta 255 caracteres).
- Datos derivados de las etapas: Se calcula de forma automática la salida y llegada programadas del vuelo, tomando como referencia la primera y la última etapa, respectivamente. Asimismo, se obtiene la ciudad de origen y destino a partir de los aeropuertos asociados a dichas etapas, y se determina el número de escalas (layovers) en función de la cantidad de etapas involucradas.
- Validaciones de consistencia: Se comprueba que, para un vuelo publicado (no en modo borrador), existan etapas asociadas y que ninguna de ellas se encuentre en estado de borrador. Además, se verifica que las etapas estén ordenadas cronológicamente y, en el caso de vuelos con una sola etapa, se invalida la opción de self-transfer. Adicionalmente, se valida que el aeropuerto de salida del primer tramo y el aeropuerto de llegada del último tramo sean distintos, garantizando así que el vuelo cuente con un origen y un destino coherentes.

Cabe destacar que estas validaciones de integridad global se ejecutan únicamente cuando el vuelo no está en modo borrador, lo que permite corregir inconsistencias durante la fase de edición sin restringir la flexibilidad mientras se prepara el vuelo para su publicación.

Con ello, se garantiza que la información relativa a los vuelos se gestiona de manera coherente y consistente, permitiendo una correcta integración de los datos de las etapas para reflejar los horarios y rutas reales de los vuelos.

### **Task 05: Track individual flight legs**

#### Descripción:

A flight aggregates several legs. A leg represents an individual segment of a flight, typically corresponding to layovers or connections. The system must store the following data for each leg: a unique flight number (composed of the airline's IATA code followed by four digits, unique), a scheduled departure and a scheduled arrival, a duration in hours, a status ("ON

TIME", "DELAYED", "CANCELLED", "LANDED"). Additionally, each leg must track the departure and arrival airports, as well as the aircraft that will be deployed for the journey.

#### Análisis:

En esta tarea se ha completado la implementación del seguimiento y gestión de las etapas (legs) que componen un vuelo, cumpliendo de forma integral con los requisitos establecidos. Se han incorporado las siguientes validaciones y funcionalidades:

- Datos obligatorios y formato: Se almacenan datos esenciales como el número de vuelo, la fecha y hora de salida y llegada, la duración (calculada en minutos) y el estado de la etapa, que puede ser "ON TIME", "DELAYED", "CANCELLED" o "LANDED". Se exige que el número de vuelo siga un formato específico (el código IATA de la aerolínea en tres letras seguido de cuatro dígitos).
- Validaciones de consistencia: Se comprueba que el aeropuerto de salida y el de llegada sean distintos, evitando registros erróneos. Además, se verifica que la hora de salida sea anterior a la de llegada, garantizando la coherencia temporal de la etapa.
- Unicidad del número de vuelo: Se implementa una doble validación para asegurar que el número de vuelo sea único. Esto incluye la comprobación basada en los últimos cuatro dígitos y una verificación completa del número, además de validar que el prefijo del número coincida con el código IATA de la aerolínea asociada al avión.

En resumen, esta tarea permite el correcto almacenamiento y control de cada etapa de un vuelo, asegurando que se cumplan las restricciones de formato, unicidad y coherencia en los horarios y datos asociados.

## Task 06: Generate sample test data

#### Descripción:

Produce assorted sample data to test your application informally. The data must include two manager accounts with credentials "manager1/manager1" and "manager2/manager2".

Create an additional manager account with credentials "manager3/manager3" that represents a manager with no associated data, except for his or her profile.

#### Análisis:

En esta tarea se ha completado la implementación del seguimiento y gestión de las etapas (legs) que componen un vuelo, cumpliendo de forma integral con los requisitos establecidos. Se han incorporado las siguientes validaciones y funcionalidades:

- Datos obligatorios y formato: Se almacenan datos esenciales como el número de vuelo, la fecha y hora de salida y llegada, la duración (calculada en minutos) y el estado de la etapa, que puede ser "ON TIME", "DELAYED", "CANCELLED" o "LANDED". Se exige que el número de vuelo siga un formato específico (el código IATA de la aerolínea en tres letras seguido de cuatro dígitos).

- Validaciones de consistencia: Se comprueba que el aeropuerto de salida y el de llegada sean distintos, evitando registros erróneos. Además, se verifica que la hora de salida sea anterior a la de llegada, garantizando la coherencia temporal de la etapa.
- Unicidad del número de vuelo: Se implementa una doble validación para asegurar que el número de vuelo sea único. Esto incluye la comprobación basada en los últimos cuatro dígitos y una verificación completa del número, además de validar que el prefijo del número coincida con el código IATA de la aerolínea asociada al avión.
- Consideración del estado de publicación: Aunque las validaciones individuales de cada etapa se aplican siempre para asegurar la integridad del formato y la unicidad del número, las restricciones de coherencia global que involucran la relación entre etapas (como la ordenación cronológica y la validación de que los aeropuertos del primer y último tramo sean diferentes) se integran en la validación del vuelo y se aplican únicamente cuando el vuelo se encuentra publicado (no en modo borrador). Esto permite mayor flexibilidad durante la edición, corrigiendo posibles inconsistencias antes de la publicación definitiva del vuelo.

En resumen, esta tarea permite el correcto almacenamiento y control de cada etapa de un vuelo, asegurando que se cumplan las restricciones de formato, unicidad y coherencia en los horarios y datos asociados, diferenciando claramente el comportamiento entre el modo borrador y el vuelo publicado.

## **Task 07: Provide GitHub planning link**

Descripción:

Provide a link to your planning dashboard in GitHub to review the tasks, their current status, and your schedule.

Análisis:

Intencionalmente en blanco

## **Task 15: Implement manager dashboards**

Descripción:

The system must handle manager dashboards with the following indicators:

The ranking the manager achieves based on their years of experience. The more years of experience, the higher the position in the ranking.

The number of years to retire, assuming that they retire at 65.

Ratio of on-time and delayed legs.

The most popular and least popular airports within their flights. An airport is popular as long as it has been an origin or destination for many flights.

The number of legs of their flights grouped by their status.

The average, minimum, maximum, and standard deviation of the cost of their flights.

#### Análisis:

En esta tarea se ha implementado el dashboard para managers, que consolida diversos indicadores clave para evaluar su desempeño y gestión. Se incluyen los siguientes aspectos:

- Ranking por experiencia: Se calcula la posición en el ranking basado en los años de experiencia, de forma que aquellos con más experiencia ocupan posiciones superiores.
- Años hasta la jubilación: Se determina el número de años restantes hasta la edad de jubilación (65 años), lo que permite evaluar la proximidad a este hito.
- Rendimiento de etapas: Se miden las proporciones de etapas de vuelo que han salido a tiempo y aquellas con retrasos, proporcionando un indicador de la eficiencia operativa.
- Popularidad de aeropuertos: Se identifican el aeropuerto más y el menos popular entre los vuelos gestionados, considerando la frecuencia con la que aparecen como origen o destino.
- Conteo de etapas por estado: Se agrupan y contabilizan las etapas según su estado (por ejemplo, a tiempo, retrasadas, canceladas o aterrizadas), lo que ofrece una visión detallada del desempeño en cada categoría.
- Estadísticas de coste de vuelos: Se calculan el coste promedio, el mínimo, el máximo y la desviación estándar de los vuelos asociados, permitiendo un análisis financiero de la operación.

Esta implementación proporciona una visión integral y cuantitativa del desempeño de cada manager, facilitando el seguimiento y la toma de decisiones estratégicas.

## Task 16: Design UML domain model

#### Descripción:

Produce a UML domain model regarding the information requirements.

#### Análisis:

Este diagrama UML refleja claramente las entidades y sus relaciones dentro del dominio de la aplicación:

- Manager: Representa al responsable de aerolínea y se asocia con sus atributos de identificación, años de experiencia, fecha de nacimiento, etc.

- **Flight:** Define los datos del vuelo (tag, self-transfer, coste, descripción, modo borrador, etc.) y se relaciona con uno o varios Leg (etapas) que conforman su itinerario.
- **Leg:** Cada etapa del vuelo incluye el número de vuelo, las fechas de salida y llegada, la duración, el estado (ON TIME, DELAYED, CANCELLED, LANDED), y referencias a la aeronave y aeropuertos implicados.
- **Airport:** Modela la información de cada aeropuerto, utilizada tanto como punto de partida como de destino de las etapas.
- **Aircraft:** Recoge los datos relativos a la aeronave utilizada en cada etapa (Leg).
- **LegStatus:** Enumera los posibles estados de una etapa, facilitando el control de su situación operativa.

En conjunto, este modelo UML permite representar y relacionar todas las entidades requeridas, garantizando que se cubran los requisitos de información y las dependencias necesarias para la correcta gestión de vuelos, etapas, aeropuertos, aeronaves y managers.

## **Task 26: Integrate weather data service**

### Descripción:

Description: The system must track weather conditions. A web service must be used to populate this entity with information about weather conditions. Thus, the exact data to store depends on the chosen service, and it is the students' responsibility to define them accordingly. It is also the students' responsibility to find the appropriate service; no implicit or explicit liabilities shall be covered by the University of Seville or their individual affiliates if the students contract pay-per-use services! The students are strongly advised to ensure that the service they choose is free of charge.

### Análisis:

En esta tarea se ha integrado la funcionalidad de seguimiento de condiciones meteorológicas mediante la creación de la entidad WeatherConditions, que almacena datos como temperaturas, humedad, viento y condiciones generales, asociados a un aeropuerto y una fecha determinada. Además, se ha utilizado el servicio gratuito de Visual Crossing (<https://www.visualcrossing.com/>) para obtener dichos datos. La integración se realiza de modo que, cuando se publique un leg de un vuelo, se invoca al servicio para recuperar las previsiones meteorológicas correspondientes a ambos aeropuertos involucrados (salida y llegada) en la fecha del leg. De esta forma, se garantiza que la información meteorológica esté alineada con el momento de operación del vuelo y se pueda disponer de datos actualizados y pertinentes para cada aeropuerto.

## **Task 27: Produce analysis report**

### Descripción:

Description: Produce an analysis report.



Análisis:

Intencionalmente en blanco

## **Task 28: Generate planning & progress report**

Descripción:

Produce a planning and progress report.

Análisis:

Intencionalmente en blanco

## **5. Conclusiones**

El análisis realizado ha permitido identificar y subsanar ambigüedades en los requisitos iniciales, implementando soluciones que garantizan la consistencia y robustez del sistema. Cada tarea se ha abordado de forma integral, desde la gestión de datos de managers y vuelos hasta la integración de servicios externos, como el de condiciones meteorológicas. La implementación de dashboards y la generación de datos de prueba refuerzan la capacidad de evaluación y seguimiento del sistema. En definitiva, las decisiones tomadas y las validaciones implementadas aseguran que el proyecto cumple con los objetivos planteados, ofreciendo una base sólida para su correcto funcionamiento y futuras ampliaciones.

## **6. Bibliografía**

Intencionadamente en blanco.