# Requirements – Student #1

Please, fill in the following form, make sure that you have ticked the requirements that you consider fulfilled, save this document, **and attach it in its original format (.docx)** to every deliverable. Regarding your ID, please keep only four random digits and mask the others using an asterisk. **Please, note that this document must be edited with the desktop version of Word since the web version does not properly support forms.** Attaching this document entails that you are the authors of the work delivered, you have not cheated in any way, and you have read and understood the information delivered regarding the subject, with a special emphasis on the methodological guidelines and how your work is going to be graded. Make sure that your project works well with the latest version of the development framework.

|  |
| --- |
| **Group:** C2.052 |
| **Repository:** https://github.com/BVP2455/Acme-ANS-C2 |
| **Student #1**  **ID Number:** 8022\*\*\*\*\*  **UVUS:**  WMR4374  **Name:**  Cegrí Marcos, José Luis  **Roles:**  Developer, Analyst, Operator, Tester |
| **Date:** Seville, 02/07/2025 |

# MANDATORY Deliverable D01: introduction

## Information requirements

Intentionally blank.

## Functional requirements

1. Modify the anonymous menu so that it shows an option that takes the browser to the home page of your favourite web site. The title must read as follows: “〈id-number〉: 〈surname〉, 〈name〉”, where “〈id-number〉” denotes your DNI, NIE, or passport number, “〈surname〉” denotes your surname/s, and “〈name〉” denotes your name/s.

X

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

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

Se proporciona un enlace a toda la planificación. Al menos en relación con D01 no sigue la metodología de trabajo explicada.

Se ha proporcionado un enlace al nuevo tablero en *Acme-ANS-C2\reports\C2\Student #1* en el que las tareas en los que los requisitos no se han cumplido han sido puestas como [S01] Task XX/T1 en cuya descripción se encuentran los comentarios del profesor que me ha corregido, y como [S01] Task XX/R1 en cuya descripción se encuentran los comentarios que he puesto en esta hoja de requisitos.

# MANDATORY Deliverable D02: data models

## Information requirements

1. **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.

X

1. 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 comes from the city of the airports to which first and last leg refers to, and finally, the **number of layovers**.

X

1. 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.

X

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

1. 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.

X

## Managerial requirements

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

X Mejora respecto a D01, pero sigue siendo una planificación incompleta (por ejemplo, faltan tareas específicas de los estudiantes)

Se ha proporcionado un enlace al nuevo tablero en *Acme-ANS-C2\reports\C2\Student #1* en el que las tareas en los que los requisitos no se han cumplido han sido puestas como [S01] Task XX/T1 en cuya descripción se encuentran los comentarios del profesor que me ha corregido, y como [S01] Task XX/R1 en cuya descripción se encuentran los comentarios que he puesto en esta hoja de requisitos.

# MANDATORY Deliverable D03: implementing features

## Information requirements

Intentionally blank.

## Functional requirements

1. Operations by **managers** on their **flights**:

* List the flights that they have created and show their details.
* Create, update, or delete their flights. Flights can be updated or deleted as long as they have not been published. For a flight to be published, it must have at least one leg, and all its legs must have been published.

OJO: la creación/actualizacion de un vuelo requiere de una conformación que no se indica en los requisitos:

A screenshot of a computer

AI-generated content may be incorrect.

He creado un vuelo con los detalles mostrados antes y le he añadido el tramo siguiente

A screenshot of a computer

AI-generated content may be incorrect.

Ahora publico el tramo y el vuelo:

A screenshot of a computer

AI-generated content may be incorrect.

A white and grey rectangular object

AI-generated content may be incorrect.

Consulto la URL y que aparece en el comando “show” y compruebo que el vuelo tiene como id el número 605. Ahora solicito la feature para añadir un nuevo leg a ese vuelo mediante la URL <http://localhost:8080/Acme-ANS-D04/manager/leg/create?flightId=605>. El resultado es el siguiente:

A screenshot of a computer

AI-generated content may be incorrect.

Evidentemente, es una petición ilegal y debería haber sido desautorizada. Relleno el formulario con los siguientes datos:

A screenshot of a computer

AI-generated content may be incorrect.

Ahora accedo al formulario del vuelo y veo lo siguiente:

A white and grey striped background

AI-generated content may be incorrect.

Ahora accedo a la lista de tramos y veo lo siguiente:

A screenshot of a computer

AI-generated content may be incorrect.

Es decir, que he podido hackear un vuelo ya publicado metiéndole un tramo un año antes de lo previsto.

Al intentar publicar un vuelo con dos tramos también obtengo un comportamiento extraño. Ni los requisitos ni el análisis realizado de los mismos incluyen esa restricción sobre que un vuelo que no requiera transbordo deba tener los mismos aeropuertos de llegada/salida en tramos consecutivos.

A screenshot of a computer

AI-generated content may be incorrect.

* FlightCreateService.java: he eliminado el requisito de la casilla de verificación
* FlightPublishService.java: he eliminado el requisito de la casilla de verificación y de si hay legs publicados y los aeropuertos no son consecutivos, no se puede modificar el atributo selfTransfer
* FlightUpdateService.java: he eliminado el requisito de la casilla de verificación y de si hay legs publicados y los aeropuertos no son consecutivos, no se puede modificar el atributo *selfTransfer*
* form.jsp: he eliminado la casilla de verificación del formulario

Tiempo total: 49 minutos

1. Operations by **managers** on their **legs**:

* List the legs in their flights ordered by their moments (no other sorting criteria is allowed).
* Show the details of their legs.
* Create and publish a leg.
* Update or delete a leg as long as it is not published.

Ver los problemas descritos anteriormente.

* **LegCreateService.java**: se ha solucionado el error por el cual se podía crear un tramo cuando el vuelo ya estaba publicado, ahora al intentar el hacking salta el authorise, y se ha eliminado el requisito de marcar la casilla de confirmación para crear un tramo
* **LegPublishService.java**: se ha eliminado el requisito de marcar la casilla de confirmación para publicar un tramo y el requisito de que los aeropuertos deban ser cuando el atributo *selfTransfer* sea true
* **LegUpdateService.java**: se ha eliminado el requisito de marcar la casilla de confirmación para actualizar un tramo
* **form.jsp**: he eliminado la casilla de verificación del formulario

Tiempo total: 63 minutos

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

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

X Ver comentarios en entrega anterior.

Se ha proporcionado un enlace al nuevo tablero en *Acme-ANS-C2\reports\C2\Student #1* en el que las tareas en los que los requisitos no se han cumplido han sido puestas como [S01] Task XX/T1 en cuya descripción se encuentran los comentarios del profesor que me ha corregido, y como [S01] Task XX/R1 en cuya descripción se encuentran los comentarios que he puesto en esta hoja de requisitos.

# MANDATORY Deliverable D04: formal testing

## Information requirements

1. Create appropriate indices for your entities, if required.

El índice para “flightNumber” está duplicado:

A screenshot of a computer program

AI-generated content may be incorrect.

Los índices definidos son insuficientes para consultas como las siguientes:

A close up of a computer screen

AI-generated content may be incorrect.

El estudiante podrá fácilmente encontrar el resto de índices que pueden faltar siguiendo la metodología estudiada.

* **Leg.java**: se ha eliminado el índice repetido del atributo *flightNumber* y he añadido el resto de indices que faltaban por añadir de otras features que utilizan *Leg* aparte de las mias mediante la busqueda *FROM Leg* del IDE

Tiempo total: 16 minutos

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

1. Produce a test suite for Requirements #8 and #9.

OJO: pruebas incompletas en manager/flight/publish.

* **create2.hack (Leg)**: he hecho el intento de post hacking que se testeo anteriormente, en este caso siendo denegado
* **create2.safe (Leg)**: restricción de cuando el aeropuerto de salida y llegada es el mismo
* **create.safe (Flight)**: he tenido que volver a hacer los test de la clase *FlightCreateService.java* debido a la eliminación de la casilla de confirmación
* **create.safe (Leg)**: he tenido que volver a hacer los test de la clase *LegCreateService.java* debido a la eliminación de la casilla de confirmación
* **FlightUpdateService.java**: he borrado algunas líneas de código que ya no se utilizan
* **LegPublishService.java**: he borrado un bucle innecesario que provocaba que la cobertura fuera menor
* **LegUpdateService.java**: he borrado una condición innecesaria de un condicional
* **publish2.safe (Leg)**: olvide verificar la restricción de no publicar cuando los tramos se solapan
* **publish.safe (Flight)**: he tenido que volver a hacer los test de la clase *FlightPublishService.java* debido a la eliminación de la casilla de confirmación
* **publish.safe (Leg)**: he tenido que volver a hacer los test de la clase *LegPublishService.java* debido a la eliminación de la casilla de confirmación
* **update2.safe (Leg)**: he metido cuando un *flightNumber* ya está en uso
* **update.safe (Flight)**: he tenido que volver a hacer los test de la clase *FlightUpdateService.java* debido a la eliminación de la casilla de confirmación
* **update.safe (Leg)**: he tenido que volver a hacer los test de la clase *LegUpdateService.java* debido a la eliminación de la casilla de confirmación

Tiempo total: 5h 37min

## Managerial requirements

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

Ver problemas en entrega anterior.

Se ha proporcionado un enlace al nuevo tablero en *Acme-ANS-C2\reports\C2\Student #1* en el que las tareas en los que los requisitos no se han cumplido han sido puestas como [S01] Task XX/T1 en cuya descripción se encuentran los comentarios del profesor que me ha corregido, y como [S01] Task XX/R1 en cuya descripción se encuentran los comentarios que he puesto en esta hoja de requisitos.

1. Produce a testing report.

El informe simplemente indica lo siguiente:

A screenshot of a document

AI-generated content may be incorrect.

Pero no se da ninguna explicación que permita explicar la conclusión sobre la base de estos resultados.

También indica que:

A screenshot of a document

AI-generated content may be incorrect.

Pero sin ningún tipo de análisis estadístico que permita determinar si existen diferencias significativas o no en los resultados.

He modificado el testing report explicando las gráficas que se adjuntan y con los nuevos resultados de la suite de pruebas al refactorizar el código, con una cobertura del 100%

Ficheros añadidos:

* **graphic.xlsx**: gráfica de la media de tiempos por cada prueba formal
* **index-graphic.xlsx**: gráfica de la media de tiempos por cada prueba formal con los índices implementados
* **older-pc-graphic.xlsx**: gráfica de la media de tiempos por cada prueba formal en otro ordenador
* **older-pc-index-graphic.xlsx**: gráfica de la media de tiempos por cada prueba formal con los índices implementados en otro ordenador
* **hipothesis-contrast.xlsx**: análisis de estadística descriptiva y z-test entre índices implementados y sin implementar
* **older-pc-hipothesis-contrast.xlsx**: análisis de estadística descriptiva y z-test entre índices implementados y sin implementar en otro ordenador
* **tester-performance.csv**: valores separados por coma del replay de la suite de pruebas formales
* **index-tester-performance.csv**: valores separados por coma del replay de la suite de pruebas formales con los índices implementados
* **older-pc-tester-performance.csv**: valores separados por coma del replay de la suite de pruebas formales en otro ordenador
* **older-pc-index-tester-performance.csv**: valores separados por coma del replay de la suite de pruebas formales con los índices implementados en otro ordenador
* **tester-performance-clean.xlsx**: libro de excel con los valores del .csv limpios
* **index-tester-performance-clean.xlsx**: libro de excel con los valores del .csv limpios con los índices implementados
* **older-pc-tester-performance-clean.xlsx**: libro de excel con los valores del .csv limpios en otro ordenador
* **older-pc-index-tester-performance-clean.xlsx**: libro de excel con los valores del .csv limpios con los índices implementados en otro ordenador

Tiempo total: 6h 05min

# SUPPLEMENTARY I Deliverable D01: introduction

## Information requirements

Intentionally blank.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

Intentionally blank.

# SUPPLEMENTARY I Deliverable D02: data models

## Information requirements

1. 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 less 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.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Produce a UML domain model regarding the information requirements.

# SUPPLEMENTARY I Deliverable D03: implementing features

## Information requirements

Intentionally blank.

## Functional requirements

1. Operations by **anonymous principals** on **user accounts**:

* Sign up to the system and become a manager.

1. Operations by **managers** on **user** **accounts**:

* Update their profiles.

1. Operations by **any principals** on **flights**:

* List the flights in the system that are published.
* Show the details of the flights that they can list (including their legs).

1. Operations by **managers** on **manager dashboards**:

* Show their manager dashboards.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Provide a link to a video in which you informally test Requirements #8 and #9. Videos should not exceed 10 minutes in length and must be stored at the USE's facilities.

# SUPPLEMENTARY I Deliverable D04: formal testing

## Information requirements

Intentionally blank.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

1. Perform five mutations in your code and report on the results.

## Managerial requirements

1. Produce a lint report.

# SUPPLEMENTARY II Deliverable D01: introduction

## Information requirements

Intentionally blank.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Produce an analysis report.

1. Produce a planning and progress report.

# SUPPLEMENTARY II Deliverable D02: data models

## Information requirements

1. 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.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Produce an analysis report.

1. Produce a planning and progress report.

# SUPPLEMENTARY II Deliverable D03: implementing features

## Information requirements

Intentionally blank.

## Functional requirements

1. Operations by **any principals** on **weather conditions**:

* List the forecast weather conditions and show their details.
* List the flights that took place in the last month under bad weather conditions.

1. Operations by **administrators** on **weather conditions**:

* Populate the database with forecast weather conditions.

## Non-functional requirements

Intentionally blank.

## Testing requirements

Intentionally blank.

## Managerial requirements

1. Produce an analysis report.

1. Produce a planning and progress report.

# SUPPLEMENTARY II Deliverable D04: formal testing

## Information requirements

Intentionally blank.

## Functional requirements

Intentionally blank.

## Non-functional requirements

Intentionally blank.

## Testing requirements

1. Produce as a complete test suite as possible for Requirement #29 ensuring that the API is properly mocked.

## Managerial requirements

1. Produce an analysis report.

1. Produce a planning and progress report.