

Universidad de Sevilla  
Escuela Técnica Superior de Ingeniería  
Informática

## Lint Report



Grado en Ingeniería Informática – Ingeniería del  
Software Diseño y Pruebas 2.

Curso 2024 – 2025

Group:C2.047

Repository: <https://github.com/HectorGuePra/C2-047-Acme-Ans-C2.git>

Student #1

ID Number: 7\*\*4\*\*7T

UVUS: joaborleo

Name: Borja León, Joaquín

Roles: Desarrollador, Analista, Tester

Date: Jul 03, 2025

-Este documento se centra en analizar los “bad smells” del código desarrollado identificados por SonarLint.

-Este informe se centrará en el código desarrollado para las features de Flight y Leg.

**ManagerFlightController:**



## Diseño y Pruebas 2 Lint Report

ManagerFlightController.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
ManagerFlightController.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
ManagerFlightController.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
ManagerFlightController.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
ManagerFlightController.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
ManagerFlightController.java	2 m...	⬆	Remove this field injection and use constructor injection instead.

- Salen avisos de inyecciones de campo debido a los @Autowired usados para registrar los servicios.

### **ManagerFlightListService:**

ManagerFlightListService.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
-------------------------------	--------	---	--

- Misma razón que los anteriores.

### **ManagerFlightShowService:**

ManagerFlightShowService.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
-------------------------------	--------	---	--

- Misma razón que los anteriores.

### **ManagerFlightCreateService:**

ManagerFlightCreateService.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
---------------------------------	--------	---	--

- Misma razón que los anteriores.


### **ManagerFlightUpdateService:**

ManagerFlightUpdateService.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
---------------------------------	--------	---	--

- Misma razón que los anteriores.


### **ManagerFlightDeleteService:**

ManagerFlightDeleteService.java	1 m...	✓	Use isEmpty() to check whether the collection is empty or not.
ManagerFlightDeleteService.java	2 m...	⬆	Remove this field injection and use constructor injection instead.
ManagerFlightDeleteService.java	2 m...	⬆	Remove this field injection and use constructor injection instead.

	Diseño y Pruebas 2 Lint Report

- Aunque SonarLint recomienda isEmpty(), se mantiene size() > 0 por preferencia personal, ya que la intención sigue siendo clara y no afecta al comportamiento del código. El resto de los avisos son iguales que los anteriores.

### **ManagerFlightPublishService:**

ManagerFlightPublishService.jav	2 m...	 Remove this field injection and use constructor injection instead.
---------------------------------	--------	--







- Misma razón que los anteriores.

### **ManagerFlightRepository:**

Resource	Date	Description	




- Ningún aviso

### **ManagerLegController:**


ManagerLegController.java	2 m...	 Remove this field injection and use constructor injection instead.
ManagerLegController.java	2 m...	 Remove this field injection and use constructor injection instead.
ManagerLegController.java	2 m...	 Remove this field injection and use constructor injection instead.
ManagerLegController.java	2 m...	 Remove this field injection and use constructor injection instead.
ManagerLegController.java	2 m...	 Remove this field injection and use constructor injection instead.
ManagerLegController.java	2 m...	 Remove this field injection and use constructor injection instead.

- Misma razón que los anteriores.

### **ManagerLegListService:**

ManagerLegListService.java	1 m...	 Define a constant instead of duplicating this literal "flightId" 5 time...
ManagerLegListService.java	1 m...	 Remove this field injection and use constructor injection instead.
ManagerLegListService.java	2 m...	 Remove this field injection and use constructor injection instead.

- Hay un aviso para usar variables en vez de repetir el literal flightId 5 veces. El resto de los avisos son iguales que los anteriores.

	Diseño y Pruebas 2 Lint Report
---	-----------------------------------

### **ManagerLegShowService:**

ManagerLegShowService.java	1 m...	Define and throw a dedicated exception instead of using a generic ...
ManagerLegShowService.java	2 m...	Remove this field injection and use constructor injection instead.
ManagerLegShowService.java	2 m...	Remove this field injection and use constructor injection instead.

- Aunque SonarLint sugiere lanzar una excepción más específica, se ha utilizado RuntimeException por simplicidad. El resto de los avisos son iguales que los anteriores.

### **ManagerLegCreateService:**

ManagerLegCreateService.java	5 da...	Correct one of the identical sub-expressions on both sides of oper...
ManagerLegCreateService.java	5 da...	Correct one of the identical sub-expressions on both sides of equal...
ManagerLegCreateService.java	1 m...	Define a constant instead of duplicating this literal "aircraft" 4 times...
ManagerLegCreateService.java	1 m...	Define a constant instead of duplicating this literal "arrivalAirport" 3...
ManagerLegCreateService.java	1 m...	Define a constant instead of duplicating this literal "departureAirpo...
ManagerLegCreateService.java	1 m...	Define a constant instead of duplicating this literal "flightId" 4 time...
ManagerLegCreateService.java	1 m...	Refactor this method to reduce its Cognitive Complexity from 32 t...
ManagerLegCreateService.java	1 m...	Define a constant instead of duplicating this literal "flightNumber" ...
ManagerLegCreateService.java	1 m...	Define a constant instead of duplicating this literal "scheduledArriv...
ManagerLegCreateService.java	1 m...	Define a constant instead of duplicating this literal "scheduledDepa...
ManagerLegCreateService.java	2 m...	Remove this field injection and use constructor injection instead.
ManagerLegCreateService.java	2 m...	Remove this field injection and use constructor injection instead.


- Hay un aviso sobre múltiples condiciones, pero debido a la simplicidad de estas no ha sido necesario modificarlo. El resto de los avisos son iguales que los anteriores.

### **ManagerLegUpdateService:**

ManagerLegUpdateService.java	1 m...	Define a constant instead of duplicating this literal "aircraft" 5 times...
ManagerLegUpdateService.java	1 m...	Define a constant instead of duplicating this literal "arrivalAirport" 4...
ManagerLegUpdateService.java	1 m...	Define a constant instead of duplicating this literal "departureAirpo...
ManagerLegUpdateService.java	1 m...	Refactor this method to reduce its Cognitive Complexity from 43 t...
ManagerLegUpdateService.java	2 m...	Remove this field injection and use constructor injection instead.

- Misma razón que los anteriores.

### **ManagerLegDeleteService:**

	Diseño y Pruebas 2 Lint Report
---	-----------------------------------

ManagerLegDeleteService.java	1 m...	Define a constant instead of duplicating this literal "aircraft" 4 times...
ManagerLegDeleteService.java	1 m...	Define a constant instead of duplicating this literal "arrivalAirport" 4...
ManagerLegDeleteService.java	1 m...	Define a constant instead of duplicating this literal "departureAirpo...
ManagerLegDeleteService.java	1 m...	Refactor this method to reduce its Cognitive Complexity from 43 t...
ManagerLegDeleteService.java	2 m...	Remove this empty statement.
ManagerLegDeleteService.java	2 m...	Remove this field injection and use constructor injection instead.
ManagerLegDeleteService.java	2 m...	Add a nested comment explaining why this method is empty, thro...

- Misma razón que los anteriores excepto el nested porque tengo el validador vacío.

### ManagerLegPublishService:

ManagerLegPublishService.java	1 m...	Define a constant instead of duplicating this literal "aircraft" 4 times...
ManagerLegPublishService.java	1 m...	Define a constant instead of duplicating this literal "arrivalAirport" 4...
ManagerLegPublishService.java	1 m...	Define a constant instead of duplicating this literal "departureAirpo...
ManagerLegPublishService.java	1 m...	Refactor this method to reduce its Cognitive Complexity from 43 t...
ManagerLegPublishService.java	2 m...	Remove this field injection and use constructor injection instead.
ManagerLegPublishService.java	2 m...	Add a nested comment explaining why this method is empty, thro...

- Misma razón que los anteriores.

### ManagerLegRepository:


- Sin avisos.

### Conclusiones:

Después de escanear las clases Java en busca de *code smells* con SonarLint, no han aparecido problemas críticos que afecten a la funcionalidad del sistema. Los avisos más frecuentes se relacionan con el uso de @Autowired en campos en lugar de inyección por constructor, el uso de size() > 0 en lugar de isEmpty(), el uso de RuntimeException genérico y una comparación incorrecta entre expresiones idénticas. También se detecta una complejidad cognitiva elevada en un método, aunque responde a un flujo de autorización con múltiples validaciones encadenadas. En general, se han corregido aquellos avisos que sí afectaban a la claridad o lógica del programa, mientras que otros se han justificado por motivos de simplicidad, estilo académico o coherencia con el resto del proyecto.