

# Testing Report

## C1.024



<https://github.com/C1-024/Acme-ANS-D04>

Jesús Martín Vergara: jesmarver@alum.us.es  
26/05/2025

	Diseño y Pruebas 2
	Testing Report


# TABLA DE CONTENIDOS

1.	RESUMEN EJECUTIVO .....	3
2.	TABLA DE REVISIÓN .....	4
3.	INTRODUCCIÓN.....	5
4.	CONTENIDO .....	6
5.	CONCLUSIONES .....	10
6.	BIBLIOGRAFÍA.....	11

	Diseño y Pruebas 2
	Testing Report


## 1. RESUMEN EJECUTIVO

Este informe documenta pruebas funcionales realizadas para características específicas del proyecto Acme ANS, centradas en operaciones con reclamaciones y registros de seguimiento. El objetivo de estas pruebas es garantizar el funcionamiento y detectar errores antes de la entrega del producto al cliente.

 <b>Acme-ANS</b>	Diseño y Pruebas 2
	Testing Report

## 2. TABLA DE REVISIÓN

Fecha	Versión	Descripción
26/05/2025	1.0	Versión inicial del testing report


	Diseño y Pruebas 2
	Testing Report

### 3. INTRODUCCIÓN

Este informe describe las pruebas funcionales y de rendimiento realizadas sobre las funcionalidades desarrolladas por el estudiante 4 (relacionadas con reclamaciones y registros de seguimiento). Para ello se han realizado dos tipos de pruebas: funcionales y de rendimiento.

Las pruebas funcionales buscaron detectar errores y asegurar medidas de seguridad adecuadas.

Las pruebas de rendimiento muestran el rendimiento del proyecto al ejecutarse en un ordenador personal para así asegurar que los tiempos de respuesta sean cortos y nuestro proyecto funcione de la manera más fluida posible.

	Diseño y Pruebas 2
	Testing Report

## 4. CONTENIDO

### 4.1. Pruebas funcionales

En este apartado nos encontramos todas las pruebas realizadas para las entidades y características del estudiante 4. Estas se dividen en Claim y Tracking Logs las cuales contienen pruebas de tipo “Safe” y pruebas de tipo “Hack”. Las pruebas de tipo “Safe” son pruebas utilizando acciones legales sobre nuestras características al contrario que las pruebas de tipo “Hack” las cuales contienen acciones ilegales para intentar descubrir vulnerabilidades en nuestro código.

#### Claim

Aquí se encuentran las pruebas relacionadas con la entidad Claim y sus características asociadas las cuales son las siguientes:

##### List-Mine-Completed


- Safe: Se ha probado el listado de las reclamaciones completadas para comprobar que todas se muestren correctamente.
- Hack: Se ha probado el listado de las reclamaciones completadas utilizando roles distintos a AssistanceAgent para comprobar que ningún usuario que no tenga ese rol pueda acceder al listado de sus reclamaciones completadas.

##### List-Mine-Undergoing

- Safe: Se ha probado el listado de las reclamaciones pendientes para comprobar que todas se muestren correctamente.
- Hack: Se ha probado el listado de las reclamaciones pendientes utilizando roles distintos a AssistanceAgent para comprobar que ningún usuario que no tenga ese rol pueda acceder al listado de sus reclamaciones pendientes.

##### Show

- Safe: Se ha probado la muestra del detalle de la reclamación para varias reclamaciones.
- Hack: Se ha probado la muestra del detalle de una reclamación de un cliente al cual no le pertenece para comprobar que no se muestre. También se ha probado acceder a dichos detalles mediante distintos roles diferentes a AssistanceAgent.

	Diseño y Pruebas 2
	Testing Report

#### Create

- Safe: Se ha probado la creación de una reclamación variando los valores de sus propiedades para comprobar que la validación se ejecute correctamente.
- Hack: Se ha probado crear una reclamación con usuarios de distinto rol a AssistanceAgent. También se ha probado la creación de una reclamación cambiando el atributo registrationMoment mediante el inspeccionar elemento del navegador para comprobar que los datos restringidos no puedan mutar.

#### Update

- Safe: Se ha probado la actualización de una reclamación variando los valores de sus propiedades para comprobar que la validación se ejecute correctamente.
- Hack: Se ha probado la actualización de una reclamación cambiando el atributo registrationMoment mediante el inspeccionar elemento del navegador para comprobar que los datos restringidos no puedan mutar. También se ha probado a actualizar una reclamación que ya está publicada comprobando que todo funcione correctamente. Por último se ha probado a llamar al endpoint de actualizar reclamación con distintos roles dando este un error de autorización.

#### Delete

- Safe: Se ha probado el borrado de reclamaciones.
- Hack: Se ha probado el uso del endpoint de borrado de reclamaciones con distintos roles. También se ha probado el borrado de una reclamación publicada para verificar que este de un error.

#### Publish

- Safe: Se ha probado la publicación de reclamaciones con distintos datos de prueba para comprobar que la validación del objeto se ejecute correctamente.
- Hack: Se ha probado a publicar una reclamación ya publicada. También se ha probado a variar el atributo registrationMoment mediante el inspeccionar elemento para verificar que dicho valor sea inmutable. Por último se ha probado llamar el endpoint de publicar reclamación mediante distintos roles a AssistanceAgent.

#### Tracking Log

Aquí se encuentran las pruebas relacionadas con la entidad TrackingLog y sus características asociadas las cuales son las siguientes:

	Diseño y Pruebas 2
	Testing Report

#### List

- Safe: Se ha probado el listado de los TrackingLogs para comprobar que todas se muestren correctamente.
- Hack: Se ha probado el listado de los TrackingLogs utilizando roles distintos a AssistanceAgent para comprobar que ningún usuario que no tenga ese rol pueda acceder al dicho listado. También se ha probado a acceder a las TrackingLogs de una reclamación que no pertenece al usuario logueado al igual que usando roles distintos a AssistanceAgent.

#### Show

- Safe: Se ha probado la muestra del detalle de algunos TrackingLogs para comprobar que se muestre correctamente.
- Hack: Se ha probado la muestra del detalle de un TrackingLog de un cliente al cual no le pertenece para comprobar que no se muestre. También se ha probado acceder a dichos detalles mediante distintos roles diferentes a AssistanceAgent.

#### Create

- Safe: Se ha probado la creación de un TrackingLog variando los valores de sus propiedades para comprobar que la validación se ejecute correctamente.
- Hack: Se ha probado crear un TrackingLog con usuarios de distinto rol a AssistanceAgent. También se ha probado la creación de un TrackingLog cambiando el atributo updateMoment mediante el inspeccionar elemento del navegador para comprobar que los datos restringidos no puedan mutar. Por último, se ha probado a crear un TrackingLog para una reclamación que no le pertenece al usuario logueado.

#### Update

- Safe: Se ha probado la actualización de un TrackingLog variando los valores de sus propiedades para comprobar que la validación se ejecute correctamente.
- Hack: Se ha probado la actualización de un TrackingLog cambiando el atributo updateMoment mediante el inspeccionar elemento del navegador para comprobar que los datos restringidos no puedan mutar. También se ha probado a actualizar un TrackingLog que ya está publicado comprobando que todo funcione correctamente. Por último se ha probado a llamar al endpoint de actualizar TrackingLog con distintos roles dando este un error de autorización.

#### Delete

- Safe: Se ha probado el borrado de TrackingLogs.





































- Hack: Se ha probado el uso del endpoint de borrado de TrackingLogs con distintos roles. También se ha probado el borrado de un TrackingLog publicado para verificar que este de un error.

#### Publish

- Safe: Se ha probado la publicación de un TrackingLog con distintos datos de prueba para comprobar que la validación del objeto se ejecute correctamente.
- Hack: Se ha probado a publicar un TrackingLog ya publicado. También se ha probado a variar el atributo updateMoment mediante el inspeccionar elemento para verificar que dicho valor sea inmutable. Por último se ha probado llamar el endpoint de publicar TrackingLogs mediante distintos roles a AssistanceAgent.

Con todas estas pruebas descritas obtenemos la siguiente cobertura de código:

▼  acme.features.assistanceAgent.claim		92,4 %	1.015	83	1.098
>  AssistanceAgentClaimController.java		100,0 %	42	0	42
>  AssistanceAgentClaimCreateService.java		98,8 %	161	2	163
>  AssistanceAgentClaimDeleteService.java		60,5 %	112	73	185
>  AssistanceAgentClaimListMineComple		99,1 %	106	1	107
>  AssistanceAgentClaimListMineUndergo		99,0 %	102	1	103
>  AssistanceAgentClaimPublishService.java		98,4 %	182	3	185
>  AssistanceAgentClaimShowService.java		100,0 %	131	0	131
>  AssistanceAgentClaimUpdateService.java		98,4 %	179	3	182
▼  acme.features.assistanceAgent.trackingLog		90,8 %	802	81	883
>  AssistanceAgentTrackingLogController.j		100,0 %	35	0	35
>  AssistanceAgentTrackingLogCreateServi		100,0 %	156	0	156
>  AssistanceAgentTrackingLogDeleteServi		60,5 %	95	62	157
>  AssistanceAgentTrackingLogListService,		100,0 %	106	0	106
>  AssistanceAgentTrackingLogPublishServ		94,2 %	162	10	172
>  AssistanceAgentTrackingLogShowService		100,0 %	106	0	106
>  AssistanceAgentTrackingLogUpdateServ		94,0 %	142	9	151

Aquí podemos ver las instrucciones que se han ejecutado al correr nuestras pruebas y como podemos ver, ambas características para Claims y TrackingLogs tienen una cobertura superior al 90% lo cual nos da confianza en nuestro código.

## 4.2. Pruebas de rendimiento

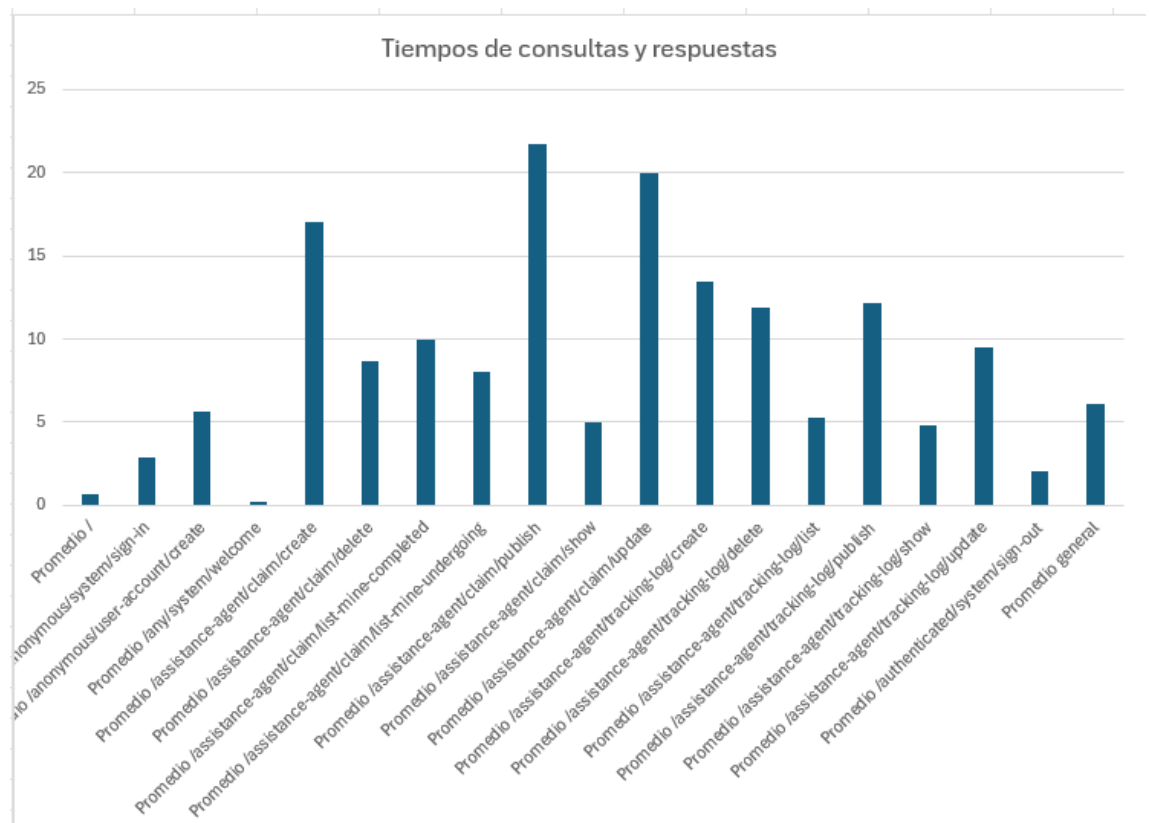
Este apartado muestra los resultados del análisis de rendimiento del sistema en mi computadora, utilizando gráficos para visualizar los tiempos de respuesta por tipo de solicitud, y comprobando que los resultados sean los óptimos.

Para ello se ha utilizado un ordenador Windows con las siguientes características:

- CPU: AMD Ryzen 5 3500X 6 core 3,60 GHz
- RAM: 16 GB DDR4 3200 GHz
- Disco: SanDisk SDSSDH3500G 500 GB


- GPU: NVIDIA GeForce GTX 1050 Ti 4 GB

Con estas características se ha obtenido la siguiente gráfica de tiempos de respuesta y consultas a nuestra base de datos:



Como podemos apreciar, los valores más altos pertenecen a las características de nuestro código que más tiempo necesita para ejecutarse. Como podemos apreciar, estos tiempos corresponden a las características “claim/publish”, “claim/update” y “claim/create”. Estas características son llamadas MIR (Most Inefficient Request). Para mejorar estos tiempos de consulta, se han implementado índices en nuestro código que mejora los tiempos de consulta con nuestra base de datos.

Ahora vamos a analizar los datos extraídos y calcular el nivel de confianza para nuestras pruebas.

 <b>Acme-ANS</b>	Diseño y Pruebas 2
	Testing Report

Columna1				
			Interval (ms)	6,25319032
			Interval (s)	0,00625319
Media	6,78438104			6,25319032
Error típico	0,27057131			0,00625319
Mediana	4,80404208			
Moda	0,2967			
Desviación estándar	7,31543954			
Varianza de la muestra	53,5156556			
Curtosis	4,16085415			
Coefficiente de asimetría	1,75431158			
Rango	48,0484			
Mínimo	0,1197			
Máximo	48,1681			
Suma	4959,38254			
Cuenta	731			
Nivel de confianza(95,0%)	0,53119072			

Como se puede ver en la imagen, el nivel de confianza del 95% es de 0,531 lo cual nos indica que podemos estar un 95% seguro de que la media real del tiempo está entre 6.253 s y 7.315 s aproximadamente.

También podemos observar una media del tiempo promedio de nuestras pruebas el cual es 6,784 s.

Como conclusión podemos afirmar que:

- La mayoría de los tiempos están agrupados en valores bajos (por debajo de la media).
- La muestra es grande, lo que da buena validez estadística.

## 5. CONCLUSIONES

Gracias a estas pruebas se ha podido verificar el funcionamiento correcto de nuestro sistema así como la fluidez de las consultas a la base de datos. También se han podido detectar errores antes de la entrega de nuestro producto asegurando mayor fiabilidad al cliente el cual obtendrá un producto sin vulnerabilidades y con una experiencia de usuario óptima.

	Diseño y Pruebas 2
	Testing Report

## 6. BIBLIOGRAFÍA

Intencionalmente en blanco.