# **Informe de Pruebas**

**Número de Grupo:** C1.009

**Repositorio:**https://github.com/DP2-2024-2025-C1-009/Acme-ANS-D04

## **Tabla de Contenido**

1. Resumen Ejecutivo
2. Tabla de Revisión
3. Pruebas Funcionales
   * Casos de Prueba para la Gestión de Aeropuertos (Admin)
     + Create
     + List
     + Show
     + Update
4. Pruebas de Rendimiento
   * Análisis de la Mejora

## **1. Resumen Ejecutivo**

En este informe se detallan las pruebas funcionales y de rendimiento realizadas sobre la funcionalidad de gestión de aeropuertos para usuarios administradores. El objetivo principal fue verificar que todas las funcionalidades (create, list, show, update) se comportaran según lo esperado, tanto en escenarios de uso seguro (.safe) como en intentos de manipulación o acceso no autorizado (.hack), y evaluar la rapidez del sistema bajo diferentes condiciones.

Para las pruebas funcionales, se organizaron los casos de prueba por funcionalidad, asegurando la correcta operación de la aplicación. En cuanto a las pruebas de rendimiento, se siguió una metodología para recopilar y procesar los tiempos de ejecución, incluyendo el cálculo de intervalos de confianza y una prueba de hipótesis para comparar el rendimiento en distintas configuraciones o máquinas.

En resumen, este informe presenta una visión completa del comportamiento del sistema en la gestión de aeropuertos, respaldado por datos de rendimiento y análisis de seguridad.

## 2.Tabla de revisión

| Versión | Fecha | Comentario |
| --- | --- | --- |
| 1.0 | 26/05/2025 | Primera versión |

## 3. Pruebas Funcionales

A continuación, se presenta una lista de pruebas funcionales realizadas para la gestión de aeropuertos, agrupadas por funcionalidad.

### **Casos de Prueba para la Gestión de Aeropuertos (Admin)**

**Create** (Crear Aeropuerto)

* Create.safe:
  + Descripción: Verifica la creación exitosa de un nuevo aeropuerto con datos válidos por parte de un administrador. Se prueban diferentes combinaciones de campos válidos.
* Create.hack:
  + Descripción: Simula intentos de crear aeropuertos con datos inválidos, IDs ya existentes, o manipulando la solicitud para intentar crear un aeropuerto sin permisos de administrador.

**List** (Listar Aeropuertos)

* List.safe:
  + Descripción: Comprueba que un administrador puede visualizar correctamente la lista completa de aeropuertos existentes. Se verifica la paginación y la correcta visualización de los datos.
* List.hack:
  + Descripción: Simula intentos de listar aeropuertos manipulando parámetros de la URL o filtros para intentar acceder a información restringida o alterar la visualización.

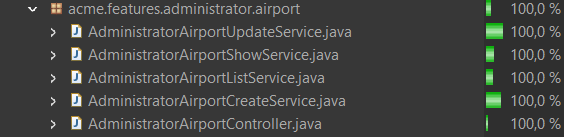
**Show** (Mostrar Aeropuerto)

* Show.safe:
  + Descripción: Garantiza que los detalles de un aeropuerto específico se muestren correctamente cuando un administrador accede a ellos.
* Show.hack:
  + Descripción: Se prueban intentos de acceso no autorizado a los detalles de aeropuertos mediante la manipulación de URLs con IDs inválidos o de aeropuertos que no existen.

**Update** (Actualizar Aeropuerto)

* Update.safe:
  + Descripción: Verifica que un aeropuerto existente pueda modificarse correctamente por un administrador utilizando datos válidos. Se prueban modificaciones parciales y completas.
* Update.hack:
  + Descripción: Comprueba que el sistema impida la modificación de aeropuertos que no existan, o que se intenten modificar campos restringidos o con datos inválidos.

La cobertura de test alcanzada de la entidad airport gestionada por administradores ha sido del 100%

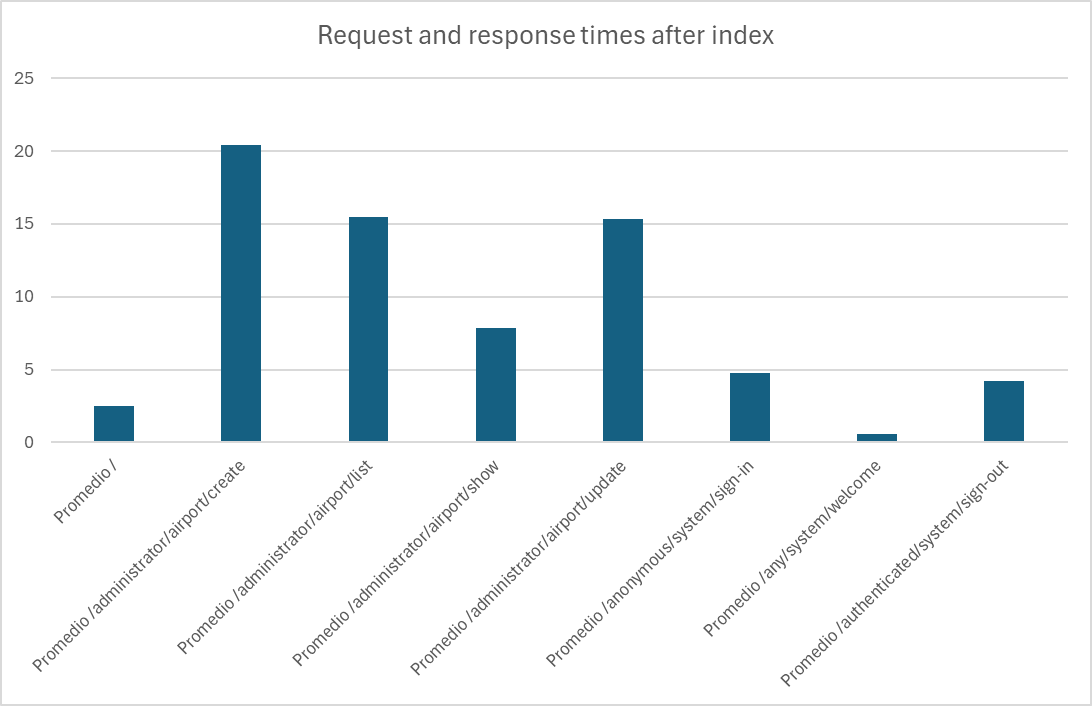
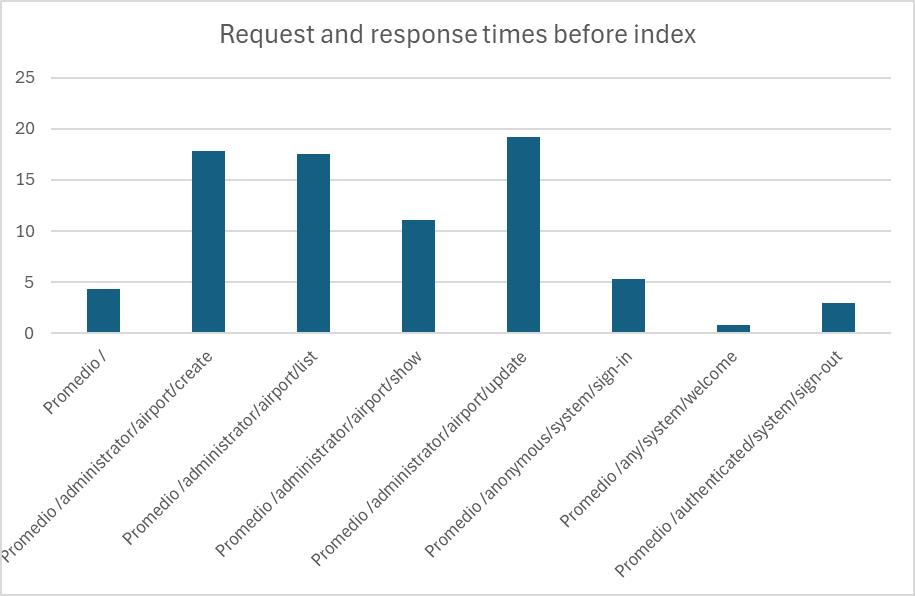


## **4. Pruebas de Rendimiento**

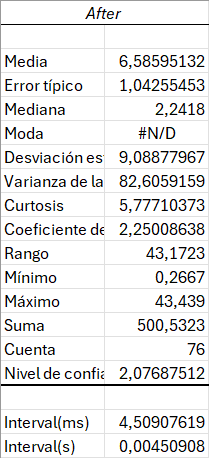
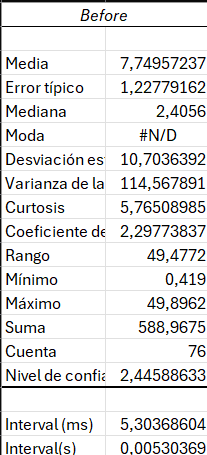
Posteriormente, se realizaron pruebas de rendimiento con el fin de optimizar el acceso a los datos y reducir los tiempos de respuesta de los servicios implicados. Para ello, se añadieron índices en los campos clave de las entidades, lo que aceleró de manera significativa las consultas en la base de datos, especialmente las operaciones de filtrado y búsqueda.

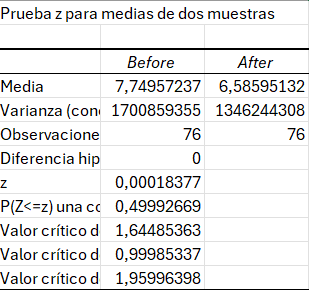
Como consecuencia, los tiempos de ejecución mejoraron de forma notable, repercutiendo positivamente en la experiencia de usuario y en la eficiencia global del sistema.

**Análisis de la mejora** Para respaldar estos resultados, se llevó a cabo un análisis estadístico en Excel mediante una prueba Z (Z-test) para comprobar si la reducción de los tiempos de respuesta tras la introducción de los índices era estadísticamente significativa. El valor obtenido, Pz < 0,00, permite rechazar la hipótesis nula y confirmar que los cambios implementados han tenido un impacto significativo y positivo en el rendimiento.



Aunque el tiempo del create ha aumentado, todos los demás han bajado con respecto al anterior lo que supone una optimización que hace que la experiencia de usuario sea mejor.





### Tras los resultados obtenidos, no hemos considerado necesario realizar un estudio del rendimiento del dispositivo frente a otro puesto que ya tenía buenos valores en el tiempo de ejecución de las pruebas y se ha realizado una comparativa.