**TESTING REPORT**

**STUDENT - 02**

* Número de grupo: C1.007
* Repositorio: <https://github.com/FernandoTC18/Acme-ANS-D04>
* Miembros del grupo:
  + Peter Philip Carter González – [percargon@alum.us.es](mailto:percargon@alum.us.es)
  + José Manuel García Rosa – [josgarros@alum.us.es](mailto:josgarros@alum.us.es)
  + Julián Romero Parejo – [julrompar@alum.us.es](mailto:julrompar@alum.us.es)
  + Samuel Tamayo Balogh – [samtambal@alum.us.es](mailto:samtambal@alum.us.es)
  + Fernando Triguero Caballo – [fertricab@alum.us.es](mailto:fertricab@alum.us.es)
* Fecha: 25/05/2025

Tabla de contenido

[Resumen ejecutivo 3](#_Toc199164413)

[Tabla de revisiones 3](#_Toc199164414)

[Functional testing 3](#_Toc199164415)

[Performance testing 7](#_Toc199164416)

[Conclusiones 9](#_Toc199164417)

## Resumen ejecutivo

Este documento presenta los resultados de las pruebas de funcionamiento realizadas en el marco del proyecto ACME-ANS de las funcionalidades requeridas al estudiante 2 del equipo de trabajo.

Durante la fase de pruebas, se evaluaron los distintos módulos correspondientes a las funcionalidades que realizaría un usuario que tenga el realm de cliente dentro de la aplicación, como la gestión de reservas, pasajeros y la relación existente entre ellos.

En conclusión, Los resultados obtenidos garantizan la correcta funcionalidad del sistema desde el punto de vista técnico.

## Tabla de revisiones

|  |  |  |
| --- | --- | --- |
| Número de revisión | Fecha | Cambios realizados |
| 1 | 25/05/2025 | Primera implementación de documento de testing. |
| 2 | 26/05/2025 | Añadido del resumen ejecutivo y del apartado de conclusiones |

## Functional testing

**Pruebas para Booking:**

* CreateBookingService:
  + Safe: Creación de un Booking con los siguientes datos:
    - Vacío.
    - Solo con LocatorCode, probando con límites y limites más y menos delta.
    - Con LocatorCode y para los dos tipos de travelClass.
    - Con LocatorCode, un travelClass y diferentes vuelos.
    - Con LocatorCode, travelClass, un vuelo y prueba con diferentes lastCardNibble, estudiando sus límites
  + Hack: Intento de creación de un booking con Flight Id 999, 183 y 197
* UpdateBookingService:
  + Safe: Actualización de un Booking con los siguientes datos:
    - Vacío.
    - Solo con LocatorCode, probando con límites y limites más y menos delta.
    - Con LocatorCode y para los dos tipos de travelClass.
    - Con LocatorCode, un travelClass y diferentes vuelos.
    - Con LocatorCode, travelClass, un vuelo y prueba con diferentes lastCardNibble, estudiando sus límites.
  + Hack: Intento de actualización de un booking con Flight Id 999, 183 y 197. Además, se ha probado para cada customer con:
    - Para customer1: 999, 373 y 379.
    - Para customer2: 999, 364 y 378.
* PublishBookingService:
  + Safe: Publicación de un Booking con los siguientes datos:
    - Vacío.
    - Solo con LocatorCode, probando con límites y limites más y menos delta.
    - Con LocatorCode y para los dos tipos de travelClass.
    - Con LocatorCode, un travelClass y diferentes vuelos.
    - Con LocatorCode, travelClass, un vuelo y prueba con diferentes lastCardNibble, estudiando sus límites.
    - Sin pasajeros.
    - Con pasajeros no publicados.
    - Con pasajeros publicados.
  + Hack: Intento de publicación de un booking con Flight Id 999, 183 y 197. Además, se ha probado para cada customer con:
    - Para customer1: 999, 373 y 379.
    - Para customer2: 999, 364 y 378.
* DeleteBookingService:
  + Safe: Eliminación de los booking con Id 365, 366, 367, 372 y 375.
  + Hack: Intento de eliminación de un booking con Flight Id 999 y 197. Además, se ha probado para cada customer con diferentes ids que pudieran provocar problemas
* ListBookingService:
  + Safe: Visualización del listado de bookings correspondientes a cada cliente.
  + Hack: Se ha intentado listar los pasajeros de los booking con booking Id:
    - Para customer1: 999, 373 y 379.
    - Para customer2: 999, 364 y 378.
* ShowBookingService:
  + Safe: visualizado de la gran parte de los ids de la base de datos desde el listado de cada cliente.
  + Hack: Intento de visualización de bookings con ids que no existen en la base de datos (999) o pertenecen a otro cliente (378 o 379 para el cliente 1 y 364 o 365 para el cliente 2).

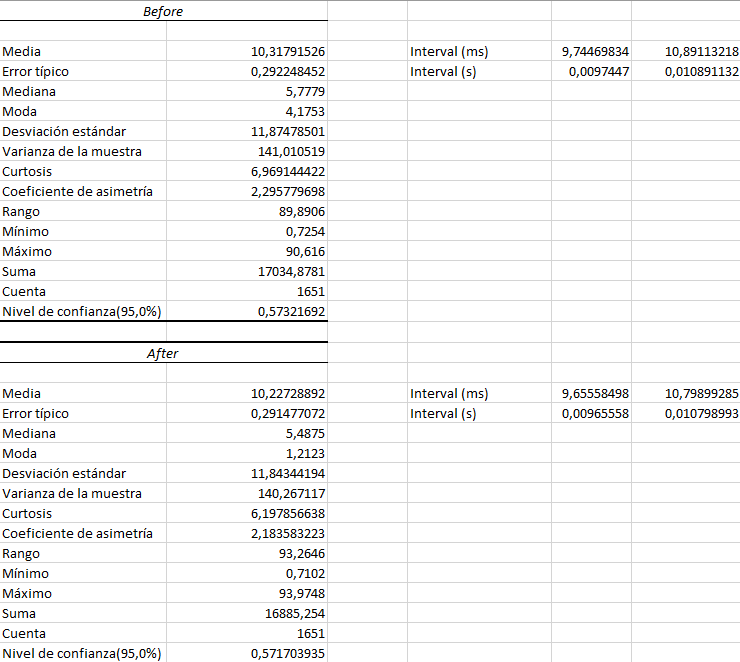
**Pruebas para Passenger:**

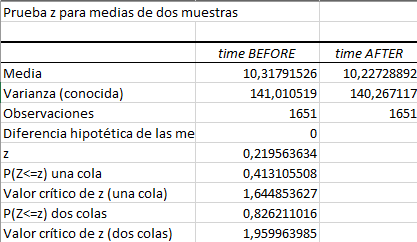
* CreatePassengerService:
  + Safe: Empleando el fichero Excel proporcionado como contenido del entorno de trabajo se ha probado la creación de un Passenger:
    - Vacío
    - Estudiando los límites del name que se compone de un máximo de 255 caracteres.
    - Con name y los emails singulares proporcionados
    - Con name, email y passport con los límites de máximo, mínimo y máximo más y menos delta y mínimo más delta
    - Con name, email, passport y birth, testeando los límites del proyecto y del atributo.
    - Con name, email, passport, birth y specialNeeds estudiando sus límites más y menos delta (máximo de 50 caracteres y opcional).
* UpdatePassengerService:
  + Safe: Empleando el fichero Excel proporcionado como contenido del entorno de trabajo se ha probado la actualización de un Passenger:
    - Vacío
    - Estudiando los límites del name que se compone de un máximo de 255 caracteres.
    - Con name y los emails singulares proporcionados
    - Con name, email y passport con los límites de máximo, mínimo y máximo más y menos delta y mínimo más delta
    - Con name, email, passport y birth, testeando los límites del proyecto y del atributo.
    - Con name, email, passport, birth y specialNeeds estudiando sus límites más y menos delta (máximo de 50 caracteres y opcional).
  + Hack: Intento de actualización de un Passenger desde cada customer con diferentes ids que deben fallar:
    - Para customer 1: 999, 414 y 403
    - Para customer 2: 999, 380 y 414
* PublishPassengerService:
  + Safe: Empleando el fichero Excel proporcionado como contenido del entorno de trabajo se ha probado la publicación de un Passenger:
    - Vacío
    - Estudiando los límites del name que se compone de un máximo de 255 caracteres.
    - Con name y los emails singulares proporcionados
    - Con name, email y passport con los límites de máximo, mínimo y máximo más y menos delta y mínimo más delta
    - Con name, email, passport y birth, testeando los límites del proyecto y del atributo.
    - Con name, email, passport, birth y specialNeeds estudiando sus límites más y menos delta (máximo de 50 caracteres y opcional).
  + Hack: Intento de publicación de un Passenger desde cada customer con diferentes ids que deben fallar:
    - Para customer 1: 999, 414 y 403
    - Para customer 2: 999, 380 y 414
* DeletePassengerService:
  + Safe: Se ha probado el borrado de Passengers con ids 389, 390, 391, 404, 405, 406 y 407.
  + Hack: Intento de borrado de un Passenger para los ids:
    - Para customer 1: 999, 414 y 403
    - Para customer 2: 999, 380 y 414
* ListFromBookingPassengerService:
  + Safe: Visualización del listado de pasajeros correspondientes a un bookingId.
  + Hack: Intento de acceder a bookingId no pertenecientes a un customer o que no se encuentran en la base de datos:
    - Para customer 1: 999, 378 y 379.
    - Para customer 2: 999, 364 y 365.
* ListPassengerService:
  + Safe: Visualización del listado de bookings correspondientes a cada cliente.
* ShowPassengerService:
  + Safe: Visualización de los pasajeros correspondientes al listado de cada uno de los clientes.
  + Hack: Intento de hacking mediante los Ids:
    - Para customer1: 999, 413 y 414.
    - Para customer2: 999, 380 y 381.

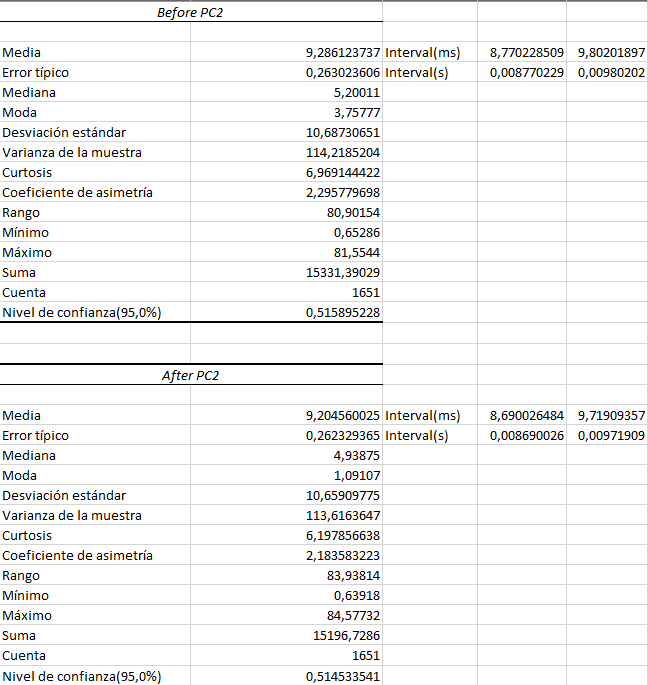
**Pruebas para BookingRecord:**

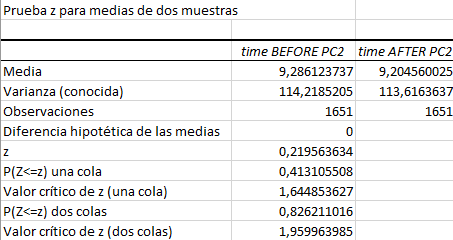
* CreateBookingRecordService:
  + Safe: Para cada uno de los clientes se han añadido diferentes Passengers a los booking (365 y 367 para el customer 1) y (375 y 367 para el customer 2)
  + Hack: Se han realizado intentos de hackeo para añadir pasajeros no permitidos a un booking:
    - Para customer 1:
      * Se ha intentado añadir los pasajeros 999, 380 y 413
      * Se ha intentado añadir un pasajero a los booking con Id 373, 999 y 379 (publicado, inexistente y de otro cliente respectivamente)
* DeleteBookingRecordService:
  + Safe: Intento de borrado de pasajeros del booking 368 y 374.
  + Hack: Intento de borrar para un Booking inexistente, publicado. O perteneciente a un Booking ya publicado, además de Passengers con diferentes ids:
    - Para customer 1: 999, 413, 403 y 380
    - Para customer 2: 999, 380, 414 y 413.

## Performance testing

****

****

****

****

Para las pruebas se han obtenido los datos del PC del estudiante y los de un PC simulado, con un mejor rendimiento. El tiempo obtenido de las distintas pruebas, así como los cálculos utilizados para determinar dichas métricas puede verse en el documento adjunto “performance-testing.xlsx” en la carpeta de reports correspondiente al estudiante 2.

## Conclusiones

Una vez finalizado el proceso de testing sobre el conjunto completo de funcionalidades, se ha alcanzado un nivel de cobertura del 99,9% para los servicios correspondientes a los módulos de Booking y Passenger, mientras que los servicios relacionados con BookingRecord han obtenido una cobertura total del 100%. Estos resultados reflejan una verificación exhaustiva del código y aseguran una elevada fiabilidad del sistema desde el punto de vista funcional.

En lo que respecta al análisis de rendimiento tras la introducción de índices, los resultados tanto en el PC del estudiante como del PC emulado (rendimiento 10% mejor) indican que no se ha producido una mejora significativa en los tiempos de respuesta. La comparación entre las métricas obtenidas antes y después de aplicar los índices muestra diferencias mínimas en los valores medios y varianzas, lo que sugiere que el rendimiento del sistema se mantiene prácticamente constante tras la optimización.

Este comportamiento queda respaldado estadísticamente mediante una prueba Z bilateral, donde el valor obtenido de P(Z ≥ z) es muy cercano a 1, indicando que la hipótesis nula —según la cual no hay diferencia significativa en el rendimiento tras la modificación— no puede ser rechazada con un nivel de confianza razonable. Por tanto, se concluye que la incorporación de índices, en este contexto particular, no introduce un impacto estadísticamente relevante sobre el rendimiento global del sistema.