**TESTING REPORT**

* **Proyecto: Acme-ANS**
* **Grupo: C0.56**
* **Repositorio:** [**https://github.com/Javclamar/Acme-ANS-D05**](https://github.com/Javclamar/Acme-ANS-D05)
* **Responsable: Javier Clavijo Martínez**
* **Fecha: 25/05/2025**

índice de contenido

[1 Resumen Ejecutivo 2](#_Toc199100198)

[2 Tabla de revisiones 2](#_Toc199100199)

[3 Testing Funcional 3](#_Toc199100200)

[3.1 Bookings 3](#_Toc199100201)

[3.2 Passengers 5](#_Toc199100202)

[3.3 PassengerRecord 7](#_Toc199100203)

[3.4 Cobertura de código 8](#_Toc199100204)

[3.5 Mutantes 9](#_Toc199100205)

[4 Performance Testing 12](#_Toc199100206)

[4.1 antes de añadir índices 12](#_Toc199100207)

[4.2 Después de añadir índices 13](#_Toc199100208)

[4.3 Conclusiones de performance testing 14](#_Toc199100209)

[5 Profiling 15](#_Toc199100210)

[5.1 Software Profiling 15](#_Toc199100211)

[5.2 Hardware Profiling 16](#_Toc199100212)

[6 Conclusiones 17](#_Toc199100213)

# Resumen Ejecutivo

Este informe representa de manera detallada el proceso de testing llevado a cabo para las features asociadas al real Customer. Se describen las funcionalidades que se han testeado, los casos de prueba que se han evaluado y los resultados obtenidos, funcionales y de rendimiento.

En el ámbito del testing funcional, para cada feature se dará un porcentaje en cobertura de código, así como los resultados de los casos de prueba utilizados.

En el ámbito de testing de rendimiento, se proporcionan análisis para cada feature que permiten comparar el rendimiento para cada una de estas.

# Tabla de revisiones

|  |  |  |
| --- | --- | --- |
| Fecha | versión | Cambios |
| 25/05/2025 | 1.0.0 | Versión Inicial |

# Testing Funcional

El testing funcional se realizó con la metodología utilizada en la asignatura, en la que se realizan pruebas positivas, negativas y de hacking. Para grabar los casos de prueba se utilizó el launcher tester-record aplicado a nuestro proyecto, y teniendo en cuenta que nop había nada que pudiera interferir con este (otro navegador, ventana, otra instancia de la aplicación…). Al grabar los casos de prueba, estos se guardan en un fichero, con extensión .safe si son casos de prueba positivos/negativos, o .hack si son casos de prueba de hacking.

Al terminar de grabar todos los casos de prueba, se utilizó el launcher tester-replay para repetirlos todos a la vez.

En esta tabla se muestran los casos de prueba para las features realizadas por un Customer sobre bookings, passengers y sus relaciones.

## Bookings

|  |  |  |
| --- | --- | --- |
| Caso de Prueba | Descripción | Resultados |
| CP-01 | Listar los bookings de un Customer, siendo Customer | Se muestra la lista |
| CP-02 | Listar los bookings sin ser Customer | Se da un fallo de autorización |
| CP-0\* | Ver los detalles de un booking publicado, siendo otro Customer | Se muestran los detalles |
| CP-03 | Ver los detalles de un booking no publicado, siendo su Customer | Se muestran los detalles |
| CP-04 | Ver los detalles de un booking, sin ser Customer | Se da un fallo de autorización |
| CP-05 | Ver los detalles de un booking no publicado, siendo otro Customer que no está relacionado | Se da un fallo de autorización |
| CP-06 | Ver los detalles de un booking, pero el id de la request es nulo | Se da un fallo de autorización |
| CP-07 | Ver los detalles de un booking, pero el id de la request da a un booking nulo | Se da un fallo de autorización |
| CP-08 | Crear un booking siendo Customer con todos los campos vacíos | Da error Not Null en los campos obligatorios |
| CP-09 | Crear/Actualizar/Publicar un booking siendo Customer, pero su localizador no esta formado por 6-8 dígitos o letras mayúsculas | Da error en el campo localizador |
| CP-10 | Crear/Actualizar/Publicar un booking siendo Customer, pero su localizador ya esta siendo usado por otro booking | Da error en el campo localizador |
| CP-11 | Crear/Actualizar/Publicar un booking siendo Customer, su localizador está formado por 6-8 dígitos o letras mayúsculas y no está siendo usado por otro booking | Se acepta el valor |
| CP-12 | Crear/Actualizar/Publicar un booking siendo Customer, con el valor ECONOMY en la clase | Se acepta el valor |
| CP-13 | Crear/Actualizar/Publicar un booking siendo Customer, con el valor BUSSINES en la clase | Se acepta el valor |
| CP-14 | Crear/Actualizar/Publicar un booking siendo Customer, con los 4 últimos dígitos de la tarjeta de crédito sin cumplir el formato de 4 enteros | Da error en el campo últimos 4 dígitos de la tarjeta usada para hacer la reserva |
| CP-15 | Crear/Actualizar/Publicar un booking siendo Customer, con un vuelo en el futuro y publicado | Se acepta el valor |
| CP-16 | Publicar un booking siendo Customer, pero no están guardados los últimos 4 dígitos de la tarjeta usada para la reserva en la base de datos | Salta un error de validación |
| CP-17 | Publicar un booking siendo Customer, pero no tiene pasajeros asociados | Salta un error de validación |
| CP-18 | Eliminar un booking siendo su Customer | Se elimina exitosamente |
| CP-19 | Publicar un booking siendo Customer, pero algún pasajero asociado no está publicado | Salta un error de validación |
| CP-20 | Crear/actualizar/publicar un vuelo con todos los campos validos | Se crea al booking |
| CP-21 | Crear un booking sin ser customer | Salta fallo de autorización |
| CP-22 | Actualizar un booking sin ser customer | Salta un fallo de autorización |
| CP-23 | Publicar un booking sin ser customer | Salta fallo de autorización |
| CP-24 | Eliminar un booking sin ser customer | Salta un fallo de autorización |
| CP-25 | Crear un booking siendo customer, pero se ha cambio el campo id del formulario | Salta un fallo de autorización |
| CP-26 | Crear/Actualizar/Publicar un booking siendo customer, pero se ha elegido un vuelo en el pasado y publicado | Salta fallo de autorización |
| CP-27 | Crear/Actualizar/Publicar un booking siendo customer, pero se ha elegido un vuelo en el pasado y no publicado | Salta un fallo de autorización |
| CP-28 | Crear/Actualizar/Publicar un booking siendo customer, pero se ha elegido un vuelo en el futuro y no publicado | Salta fallo de autorización |
| CP-29 | Crear/Actualizar/Publicar un booking siendo customer, pero se ha elegido un vuelo nulo | Salta un fallo de autorización |
| CP-30 | Actualizar/Publicar/Eliminar un Booking pero el método es GET | Salta fallo de autorización |

## Passengers

|  |  |  |
| --- | --- | --- |
| Caso de Prueba | Descripción | Resultado |
| CP-01 | Listar los passengers siendo Customer | Se muestra la lista |
| CP-02 | Listar los passengers sin ser Customer | Salta fallo de autorización |
| CP-03 | Mostrar los detalles de un Passenger siendo su Custoemr asociado | Se muestran los detalles |
| CP-04 | Mostrar los detalles de un Passenger sin ser Customer | Salta fallo de autorización |
| CP-05 | Mostrar los detalles de un Passenger sin ser su Customer asociado | Salta fallo de autorización |
| CP-06 | Mostrar los detalles de un Passenger, pero la id es nula (GET HACKING) | Salta fallo de autorización |
| CP-07 | Crear/Actualizar/Publicar un Passenger con el formulario vacío | Da errores NotNull en los valores obligatorios (todos excepto necesidades especiales) |
| CP-08 | Crear/Actualizar/Publicar un Passenger siendo su nombre 256 caracteres | Da error de validación en el campo Nombre Completo |
| CP-09 | Crear/Actualizar/Publicar un Passenger siendo su nombre 1 carácter | Se acepta el valor |
| CP-10 | Crear/Actualizar/Publicar un Passenger siendo su nombre 2 caracteres | Se acepta el valor |
| CP-11 | Crear/Actualizar/Publicar un Passenger siendo su nombre 254caracteres | Se acepta el valor |
| CP-12 | Crear/Actualizar/Publicar un Passenger siendo su nombre 255 caracteres | Se acepta el valor |
| CP-13 | Crear/Actualizar/Publicar un Passenger siendo su email válido(a@e.e/e.es/e.esx) | Se acepta el valor |
| CP-14 | Crear/Actualizar/Publicar un Passenger siendo su email inválido(user:mail@...) | Da error de validación en el campo email |
| CP-15 | Crear/Actualizar/Publicar un Passenger siendo su pasaporte una combinación de 6-9 dígitos o letras mayúsculas | Se acepta el valor |
| CP-16 | Crear/Actualizar/Publicar un Passenger siendo su pasaporte 5 dígitos | Da error de validación en el campo pasaporte |
| CP-17 | Crear/Actualizar/Publicar un Passenger siendo su pasaporte 10 dígitos | Da error de validación en el campo pasaporte |
| CP-18 | Crear/Actualizar/Publicar un Passenger siendo su pasaporte 5 letras mayúsculas | Da error de validación en el campo pasaporte |
| CP-19 | Crear/Actualizar/Publicar un Passenger siendo su pasaporte 10 letras mayúsculas | Da error de validación en el campo pasaporte |
| CP-20 | Crear/Actualizar/Publicar un Passenger con fecha de nacimiento entre 1990/01/01 00:00(moment.minimum) y 2025/01/01 00:00(base.moment) | Se acepta el valor |
| CP-21 | Crear/Actualizar/Publicar un Passenger con fecha de nacimiento menor a 1990/01/01 00:00 | Da error de validación en el campo fecha de nacimiento |
| CP-22 | Crear/Actualizar/Publicar un Passenger con fecha de nacimiento mayor a 2025/01/01 00:00 | Da error de validación en el campo fecha de nacimiento |
| CP-23 | Crear/Actualizar/Publicar un Passenger siendo sus necesidades especiales 1 carácter | Se acepta el valor |
| CP-24 | Crear/Actualizar/Publicar un Passenger siendo sus necesidades especiales 2 caracteres | Se acepta el valor |
| CP-25 | Crear/Actualizar/Publicar un Passenger siendo sus necesidades especiales 49 caracteres | Se acepta el valor |
| CP-26 | Crear un Passenger siendo Customer pero se ha modificado el campo id del formulario | Salta fallo de autorización |
| CP-27 | Actualizar/Publicar/Eliminar un Passenger sin ser Customer | Salta fallo de autorización |
| CP-28 | Actualizar/Publicar/Eliminar un Passenger sin ser su Customer asociado | Salta fallo de autorización |
| CP-29 | Actualizar/Publicar/Eliminar un Passenger siendo su Customer pero usando el método GET(GET HACKING) | Salta fallo de autorización |
| CP-30 | Actualizar/Publicar/Eliminar un Passenger siendo su Customer pero este esta publicado | Salta fallo de autorización |
| CP-31 | Listar los pasajeros de un booking siendo su Customer asociado | Se listan los pasajeros exitosamente |
| CP-32 | Listar los pasajeros de un booking sin ser Customer | Salta fallo de autorización |
| CP-33 | Listar los pasajeros de un booking sin ser su Customer asociado | Salta fallo de autorización |
| CP-34 | Listar los pasajeros de un booking pero el masterId es nulo | Salta fallo de autorización |
| CP-35 | Listar los pasajeros de un booking pero el masterId referencia a un booking nulo | Salta fallo de autorización |
| CP-36 | Listar los pasajeros de un booking pero el masterId referencia a un de otro Customer | Salta fallo de autorización |

## PassengerRecord

|  |  |  |
| --- | --- | --- |
| Caso de Prueba | Descripción | Resultados |
| CP-01 | Crear un PassengerRecord siendo el Customer asociado del booking que se referencia | Se crea la relación exitosamente |
| CP-02 | Crear un PassengerRecord sin ser Customer | Se da un fallo de autorización |
| CP-03 | Crear un PassengerRecord sin ser el Customer asociado al booking que se referencia | Se da un fallo de autorización |
| CP-04 | Crear un PassengerRecord pero el masterId es nulo | Se da un fallo de autorización |
| CP-05 | Crear un PassengerRecord pero el masterId referencia a un booking nulo | Se da un fallo de autorización |
| CP-06 | Crear un PassengerRecord pero el masterId referencia a un booking publicado | Se da un fallo de autorización |
| CP-07 | Crear un PassengerRecord pero el masterId referencia a un booking de otro Customer no publicado | Se da un fallo de autorización |
| CP-08 | Crear un PassengerRecord pero el masterId referencia a un booking de otro Customer publicado | Se da un fallo de autorización |
| CP-09 | Crear un PassengerRecord pero el id del Passenger elegido es nulo | Se da un fallo de autorización |
| CP-10 | Crear un PassengerRecord pero el Passenger elegido es nulo | Se da un fallo de autorización |
| CP-11 | Crear un PassengerRecord pero el Passenger elegido no está publicado | Se da un fallo de autorización |
| CP-12 | Crear un PassengerRecord pero el Passenger elegido ya está en la misma reserva | Se da un fallo de autorización |
| CP-13 | Crear un PassengerRecord pero el Passenger elegido es de otro Customer y no está publicado | Se da un fallo de autorización |
| CP-14 | Crear un PassengerRecord pero el Passenger elegido es de otro Customer y está publicado | Se da un fallo de autorización |
| CP-15 | Eliminar un PassengerRecord siendo el Customer asociado al booking que se referencia y el Passenger elegido pertenece a la reserva y al Customer | Se elimina la relación exitosamente |
| CP-16 | Eliminar un PassengerRecord sin ser Customer | Se da un fallo de autorización |
| CP-17 | Eliminar un PassengerRecord sin ser el Customer asociado al booking que se referencia | Se da un fallo de autorización |
| CP-19 | Eliminar un PassengerRecord pero el masterId es nulo | Se da un fallo de autorización |
| CP-20 | Eliminar un PassengerRecord pero el masterId referencia a un booking nulo | Se da un fallo de autorización |
| CP-21 | Eliminar un PassengerRecord pero el masterId referencia a un booking publicado | Se da un fallo de autorización |
| CP-22 | Eliminar un PassengerRecord pero el masterId referencia a un booking de otro Customer no publicado | Se da un fallo de autorización |
| CP-23 | Eliminar un PassengerRecord pero el masterId referencia a un booking de otro Customer publicado | Se da un fallo de autorización |
| CP-24 | Eliminar un PassengerRecord pero el id del Passenger elegido es nulo | Se da un fallo de autorización |
| CP-25 | Eliminar un PassengerRecord pero el Passenger elegido es nulo | Se da un fallo de autorización |
| CP-26 | Eliminar un PassengerRecord pero el Passenger elegido no está en la reserva referenciada | Se da un fallo de autorización |

## Cobertura de código

A screenshot of a computer

AI-generated content may be incorrect.  
  
Como podemos ver hemos conseguido con estos tests un 100% de cobertura de código en todas las features, excepto en las relacionadas a PassengerRecord, que tiene un 99,2% debido a que su PassengerRecordCreate tiene un 98,4%.

## Mutantes

En este apartado vemos como afecta la adición de mutantes en diferentes partes del proyecto(entidades y servicios) de las features relacionadas con Customer, Booking y Passenger.

* **Mutante 1: Cambio de restricción en la entidad Booking**

**A screen shot of a message

AI-generated content may be incorrect.**

A screen shot of a message

AI-generated content may be incorrect.

Se ha modificado la restricción del atributo lastCreditCardDigits, que los requisitos indican que debe ser un entero de 4 dígitos.

A screen shot of a computer code

AI-generated content may be incorrect.

Como podemos ver, ya los tests de creación de booking dan error, por lo que esta restricción esta implementada de una manera correcta.

* ******Mutante 2: Cambio de operador lógico en el servicio createBooking**

En este caso se modifica una línea del método authorise, en la que se comprueba que los vuelos introducidos son válidos. Cambiamos el operador AND por ORA screen shot of a computer code

AI-generated content may be incorrect.  
  
En este caso el error ha salido en los tests de hackeo del servicio créate, pero podemos ver que la comprobación está bien implementada, ya sea para evitar un error fatal, o un intento de hackeo

* **Mutante 3: Cambio de operador relacional en el servicio createBooking**





En este caso intercambiamos el operador ¡= 0 por >= 0

A black background with many small colored lines

AI-generated content may be incorrect.

Como podemos ver en el test suite create.safe de la feature create de booking, por lo que si el operador es diferente de ¡=0, la feature es inutilizable

* **Mutante 4: Cambio de una restricción en la entidad Passenger**

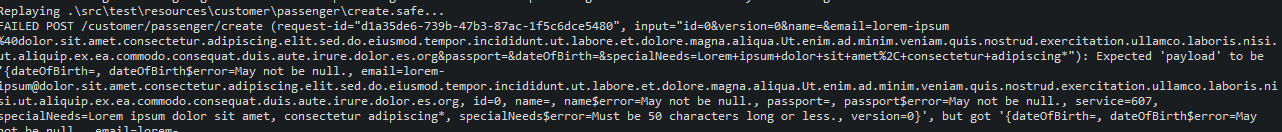
A black rectangle with white dots

AI-generated content may be incorrect.

A math equation on a black background

AI-generated content may be incorrect.

Este mutante cambia la restricción sobre el atributo specialNeeds de la entidad Passenger. La restricción @ValidShortText simplemente aplica un @ValidString con min=0 y max=50.



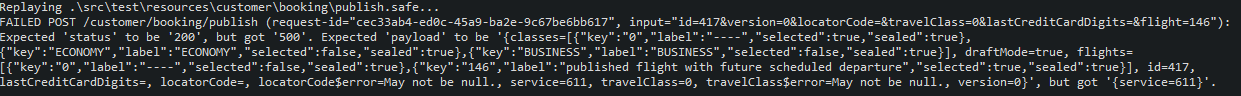
Como podemos ver, la test suite para el servicio create de Passenger da error, por lo que es una restricción implementada correctamente.

* **Mutante 5: Cambio de operador lógico en el servicio publish de Booking**





En este caso cambiamos el operador .after por .before en el método authorise, esto debe generar fallos, ya que solamente se podrían publicar bookings con vuelos que estén en el pasado.



Como se ve, la tests suite publish.suite dirigida a booking ha dado errores, lo que indica que esta bien implementada la restricción

# Performance Testing

En este apartado vermos métricas respecto al rendimiento de la aplicación, siempre relacionado a las features asociadas a Customer, antes y después de añadir índices a las tablas de la base de datos.

Estos índices se añaden en base a las consultas que se hacen desde los repositorios y los atributos que se utilizan de cada entidad.

Los siguientes tests han sido realizados con un sistema que consta con 16GB de memoria RAM y un Intel I5-10400F como procesador.

## antes de añadir índices

A screenshot of a spreadsheet

AI-generated content may be incorrect.

## Después de añadir índices

A screenshot of a spreadsheet

AI-generated content may be incorrect.

## Conclusiones de performance testing

Viendo esto, el rendimiento ha mejorado 1ms según la media, aunque la mediana ha subido. El intervalo de confianza del 95% para los tiempos después de añadir los índices es de [13,17445 – 16,32964]; por lo que se cumple que el limite superior sea a 1000ms

Además, se ha realizado un contraste de hipótesis, obteniendo un p-valor de 0,799598. Este valor es mucho mayor al umbral de significación (0,5), por lo que no hay una mejora significativa del rendimiento tras añadir índices a las tablas.

A screenshot of a computer

AI-generated content may be incorrect.

# Profiling

## Software Profiling

En este apartado se hace un profiling de los métodos relacionados con los servicios de las features de Booking, Passsenger y PassengerRecord. Esto lo conseguimos realizando los tests anteriores y utilizando VisualVM.

A screenshot of a computer

AI-generated content may be incorrect.

Viendo estos resultados, se deduce que ninguno de los métodos implementados en los servicios de las features son los que consumen tiempo, sino métodos externos o de clases de las que extienden.

## Hardware Profiling

En este apartado hacemos un hardware profiling para ver si alguno de los componentes del sistema usado para hacer los tests forman un cuello de botella.

A screen shot of a graph

AI-generated content may be incorrect.

Como podemos ver, ni la CPU(negro), ni la memoria(amarillo), ni el adaptador de red(azul) o cualquier oreo componente generan un cuello de botella. El único que ha llegado al 100% ha sido el Disco Lógico, que en un punto ha llegado al 100%, pero puede haber sido por algún proceso externo, ya que en el resto de tiempo de ejecución se ha mantenido en máximos de 20%.

# Conclusiones

Viendo los resultados de los diferentes tipos de testing, vemos que las features del Student 2 funcionan como se esperaban, tanto en funcionalidad como en rendimiento.

Funcionalmente vemos el 100% de la cobertura de código, comprobando todos los posibles casos en los servicios y viendo que se da su resultado esperado(positivo, negativo o fallo de autorización).

En los tests de rendimiento, podemos ver un pequeño aumento del rendimiento al añadir los índices a las tablas de Booking, Passenger y PassengerRecord, aunque no ha sido muy notable

A screen shot of a message

AI-generated content may be incorrect.

A screen shot of a message

AI-generated content may be incorrect.