Universidad de Sevilla  
Escuela Técnica Superior de Ingeniería Informática

**Testing Report**



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

Curso 2023 – 2024

| **Fecha** | **Versión** |
| --- | --- |
| 25/05/2024 | v1.1.0 |

| **Grupo: C1.011** | | |
| --- | --- | --- |
| **Autores por orden alfabético** | **Rol** | **Descripción del rol** |
| Abouri, Mohamed - Y7156458E | Developer | Persona encargada de desarrollar el código. |
| Cabello Ranea, Rafael -32094496C | Operador | Encargado de las tareas de campo, de las instalaciones y del mantenimiento de los sistemas de la empresa. |
| Calderón Rodríguez, Francisco Javier - 78233219F | Project Manager | Persona encargada de tomar decisiones de diseño y vigilar el correcto desarrollo |
| Delgado Pallares, David - 29519510E | Tester | Persona encargada de realizar pruebas sobre el código. |
| Ramírez Lara, Iván- 77852382G | Developer | Persona encargada de desarrollar el código. |

**Control de Versiones**

| **Fecha** | **Versión** | **Descripción** |
| --- | --- | --- |
| 24/05/2024 | v1.0.0 | Desarrollo de la primera versión. |
| 25/05/2024 | v1.1.0 | Añadidos datos de coverage y gráficas de prueba. |
|  |  |  |
|  |  |  |

**Índice de contenido**

[1. Introducción 2](#_heading=h.gjdgxs)

2[.](#_heading=h.gjdgxs) Testeo funcional 2

3[. Testeo de re](#_heading=h.30j0zll)ndimiento 10

4. Cobertura 15

1. **Introducción**

Este documento presenta dos capítulos esenciales: pruebas funcionales, con un listado de casos de prueba organizados por característica y su efectividad en detectar errores, y pruebas de rendimiento, con gráficos e intervalos de confianza del 95% para el tiempo de respuesta en dos computadoras, además de un análisis para determinar cuál es más potente. Estos capítulos son cruciales para evaluar la calidad y el rendimiento del software.

1. **Testeo funcional**

En esta sección se detallarán todas las pruebas que hemos realizado para testear el funcionamiento de nuestro software. Se explicará qué es lo que se ha hecho, el resultado esperado, el resultado real (que hace referencia al resultado del test corregido después de encontrar bugs que hicieran que el código estuviera mal) y, por último, la columna de bugs encontrados. Si ha habido bugs, se ha arreglado el test, por lo que el resultado real ya no los contiene.

Descripción de la prueba: Se detallará cada prueba realizada, incluyendo los pasos específicos llevados a cabo.

Resultado esperado: Se especificará cuál era el comportamiento o el resultado que se esperaba obtener con la prueba.

Resultado real: Se presentará el resultado obtenido tras la ejecución del test, ya corregido después de encontrar y solucionar cualquier bug que afectara al código.

Bugs encontrados: Se listarán los bugs detectados durante la prueba y las correcciones realizadas. Una vez solucionados los bugs, se actualizará el resultado real para reflejar el funcionamiento correcto del software.

**List**

| **Entidad** | **Descripción** | **Resultado Esperado** | **Resultado Real** | **Bugs detectados** |
| --- | --- | --- | --- | --- |
| **sponsorship safe** | Verificar que un sponsor puede listar sus patrocinios. | El sistema deberá mostrar los patrocinios asociados al sponsor | El sistema ha mostrado los patrocinios asociados al sponsor | Ninguno |
| **sponsorship hack** | Verificar que un rol no sponsor no puede listar patrocinios. | El sistema no debe mostrar los patrocinios, usando un rol no sponsor, dando un error de pánico | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. | Ninguno |
| **invoice safe** | Comprobar que un sponsor puede listar los recibos de los patrocinios. | El sistema deberá mostrar los recibos asociadas a un patrocinio del sponsor | El sistema ha mostrado todas los recibos asociadas a cada uno de los patrocinios probados | Ninguno |
| **invoice hack** | Comprobar que un rol no sponsor no puede listar recibos asociadas a un patrocinio determinado. | El sistema no debe mostrar los recibos asociado a un patrocinio, usando un rol no sponsor, dando un error de pánico | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. También ha dado error cuando se ha intentado acceder a las user stories de un patrocinio con un sponsor diferente. | Ninguno |

**Show**

| **Entidad** | **Descripción** | **Resultado Esperado** | **Resultado Real** | **Bugs detectados** |
| --- | --- | --- | --- | --- |
| **sponsorship safe** | Verificar que un sponsor puede ver toda la información de un patrocinio. | El sistema deberá mostrar la información de los patrocinios asociados al sponsor | El sistema ha mostrado la información de los patrocinios asociados al sponsor | Ninguno |
| **sponsorship hack** | Verificar que un rol no sponsor no puede ver la información de los patrocinios. | El sistema no debe mostrar la información de patrocinios, usando un rol no sponsor, dando un error de pánico | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. | Ninguno |
| **invoice safe** | Comprobar que un sponsor puede ver la información asociada a los recibos. | El sistema deberá mostrar la información de las user stories asociados al sponsor. | El sistema ha mostrado toda la información de los recibos elegidas y que están asociadas al sponsor | Ninguno |
| **invoice hack** | Comprobar que un rol no sponsor no puede ver la información asociada a los recibos. | El sistema no debe mostrar los recibos, usando a un rol no sponsor, dando un error de pánico | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. Al intentar acceder a la información de una recibo. Tambíen lo ha dado si la recibo no pertenece a el sponsor utilizado | Ninguno |

**Create**

| **Entidad** | **Descripción** | **Resultado Esperado** | **Resultado Real** | **Bugs detectados** |
| --- | --- | --- | --- | --- |
| **sponsorship safe** | Verificar que un sponsor puede crear patrocinios de forma correcta. | El sistema deberá permitir crear patrocinios válidos. | El sistema ha dejado crear patrocinios con datos válidos. | Se descubrió que se podía publicar un patrocinio con un código duplicado y esto causaba **error 500**. |
| **sponsorship hack** | Verificar que un rol no sponsor no puede acceder a la creación de patrocinios. | El sistema debe de lanzar un error de pánico si un rol no sponsor intenta acceder a la URL o si se hace una petición sin ser sponsor. | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. | Ninguno |
| **invoice safe** | Comprobar que un sponsor puede crear recibo de forma correcta. | El sistema deberá permitir crear recibos válidas. | El sistema ha permitido crear recibos válidas. | Ninguno |
| **invoice hack** | Revisar que un rol no sponsor no puede acceder a la creación de recibos. | El sistema debe de lanzar un error de pánico si un rol no sponsor intenta acceder a la URL o si se hace una petición sin ser sponsor. | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. Al intentar acceder a la información de una recibo. También lo ha dado si la recibo no pertenece al sponsor utilizado | Ninguno |

**Publish**

| **Entidad** | **Descripción** | **Resultado Esperado** | **Resultado Real** | **Bugs detectados** |
| --- | --- | --- | --- | --- |
| **sponsorship safe** | Verificar que un sponsor puede publicar patrocinios válidos de forma correcta. | El sistema deberá permitir publicar patrocinios válidos. | El sistema ha dejado publicar patrocinios con datos válidos. | Se descubrió que la etiqueta de error de dinero negativo no mostraba el mensaje correctamente. |
| **sponsorship hack** | Verificar que un rol no sponsor u otro sponsor no puede acceder a la publicación de patrocinios. | El sistema debe de lanzar error de pánico si un rol no sponsor intenta acceder a la URL, si se hace una petición sin ser sponsor o si se hace desde un sponsor diferente | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. También ha dado error 500 si un sponsor intentaba acceder a un patrocinio de otro sponsor | Ninguno |
| **invoice safe** | Comprobar que un sponsor puede publicar un recibo de forma correcta. | El sistema deberá permitir publicar recibos válidos. | El sistema ha permitido publicar los recibos elegidos. | Ninguno |
| **invoice hack** | Comprobar que un rol no sponsor no puede acceder a la publicación de recibos. | El sistema debe lanzar un error de pánico si un rol no sponsor intenta acceder a la URL o si se hace una petición sin ser sponsor. | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. Al intentar acceder a la información de una recibo. También lo ha dado si la recibo no pertenece al sponsor utilizado | Ninguno |

**Delete**

| **Entidad** | **Descripción** | **Resultado Esperado** | **Resultado Real** | **Bugs detectados** |
| --- | --- | --- | --- | --- |
| **sponsorship safe** | Verificar que un sponsor puede borrar patrocinios de forma correcta. | El sistema deberá permitir eliminar patrocinios que no estén publicados. | El sistema ha dejado eliminar patrocinios no publicados. | Ninguno |
| **sponsorship hack** | Verificar que un rol no sponsor no puede acceder a la eliminación de patrocinios. | El sistema debe lanzar un error de pánico si un rol no sponsor intenta acceder a la URL o si se hace una petición sin ser sponsor. | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor y en sponsor no autorizados. | Ninguno |
| **invoice safe** | Comprobar que un sponsor puede eliminar el recibo de forma correcta. | El sistema deberá permitir eliminar recibos no publicados. | El sistema ha permitido eliminar recibos válidos. | Ninguno |
| **invoice hack** | Revisar que un rol no sponsor no puede acceder a la eliminación de recibos. | El sistema debe lanzar un error de pánico si un rol no sponsor intenta acceder a la URL o si se hace una petición sin ser sponsor. | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. Al intentar acceder a la información de una recibo. También lo ha dado si la recibo no pertenece al sponsor utilizado | Ninguno |

**Update**

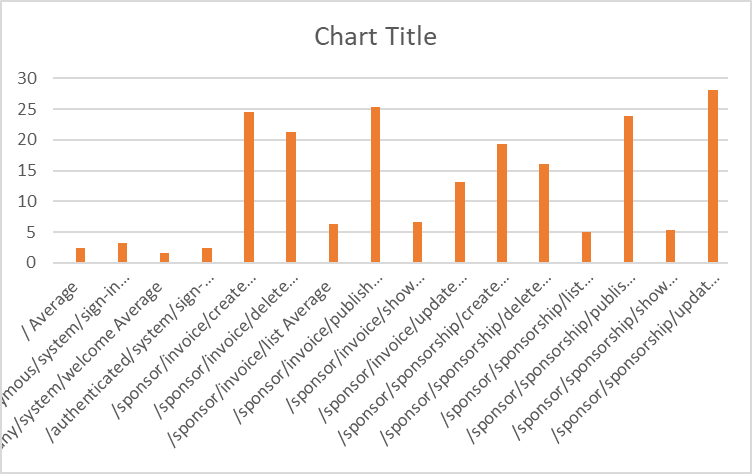
| **Entidad** | **Descripción** | **Resultado Esperado** | **Resultado Real** | **Bugs detectados** |
| --- | --- | --- | --- | --- |
| **sponsorship safe** | Verificar que un sponsor puede actualizar patrocinios válidos de forma correcta. | El sistema deberá permitir actualizar patrocinios válidos. | El sistema ha dejado actualizar patrocinios con datos válidos. | Se descubrió que la etiqueta de error no mostraba el mensaje correctamente. |
| **sponsorship hack** | Verificar que un rol no sponsor u otro sponsor no puede acceder a la actualización de patrocinios. | El sistema debe lanzar un error de pánico si un rol no sponsor intenta acceder a la URL, si se hace una petición sin ser sponsor o si se hace desde un sponsor diferente | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. También ha dado error 500 si un sponsor intentaba acceder a un patrocinio de otro sponsor | Ninguno |
| **invoice safe** | Comprobar que un sponsor puede actualizar un recibo de forma correcta. | El sistema deberá actualizar los recibos válidos. | El sistema ha permitido actualizar los recibos elegidos. | Se descubrió que la etiqueta de error no mostraba el mensaje correctamente. |
| **invoice hack** | Comprobar que un rol no sponsor no puede acceder a la actualización de recibos. | El sistema debe lanzar un error de pánico si un rol no sponsor intenta acceder a la URL o si se hace una petición sin ser sponsor. | El sistema ha dado un error 500 de no autorizado en cada rol no sponsor. Al intentar acceder a la información de una recibo. También lo ha dado si la recibo no pertenece al sponsor utilizado | Ninguno |

1. **Testeo de rendimiento**

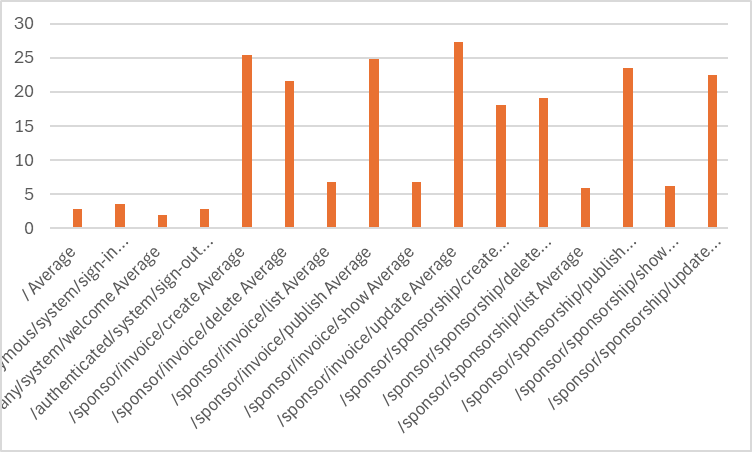
En este capítulo, se evaluará el rendimiento de nuestro patrocinio mediante pruebas detalladas que analizan el tiempo de respuesta al atender las solicitudes en nuestras pruebas funcionales. Utilizaremos dos muestras, una de ellas estará compuesta por los resultados obtenidos por hacer realplayer desde coverage, sobre las entidades con índices y sin estos. La segunda muestra se hará de la misma forma, pero desde el apartado de debug.

Para realizar el análisis de estas pruebas, presentaremos gráficos que ilustran los resultados obtenidos. Además, calcularemos intervalos de confianza del 95% para los tiempos de respuesta medidos y realizaremos un contraste de hipótesis, también con un intervalo de confianza del 95%, para determinar la diferencia de hacer realplayer sobre entidades con índices o entidades sin estos.

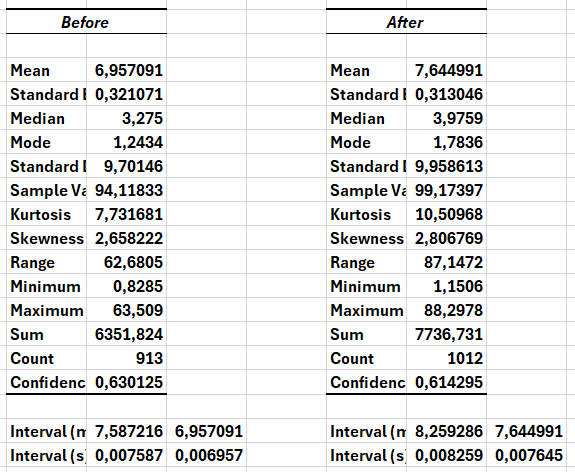
**Gráficas de entidades sin índices:**

****

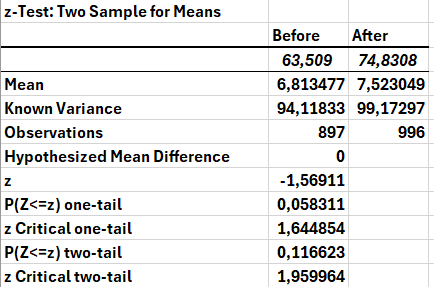
**Gráficas de entidades con índices:**

****

**Comparación de resultados:**

****

**Prueba z:**

****

En nuestras pruebas de rendimiento, hemos obtenido un valor crítico de z de **0.1166** para un nivel de significancia (α) de **0.95**. Este valor se utiliza para determinar si las diferencias en los tiempos de respuesta entre las dos pruebas son estadísticamente significativas.

De acuerdo con nuestra metodología, si el p-valor se encuentra en el intervalo (α, 1.00], esto indica que los cambios no resultaron en mejoras relevantes; es decir, aunque los tiempos de muestra sean diferentes, globalmente son equivalentes. Dado que nuestro valor crítico de z no está en este intervalo, concluimos que hay una mejora significativa en el rendimiento al comparar los tiempos de respuesta de las pruebas. En resumen, las diferencias observadas son lo suficientemente grandes como para considerarse estadísticamente relevantes con el nivel de confianza del 95%.

**Advertencia:**

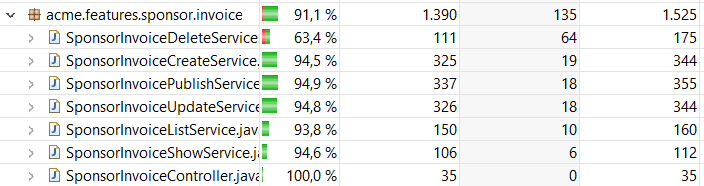
Los resultados de esta prueba han sido afectados por un bug que ocurre en el framework, ya que hay veces que cuando se ejecuta el realplayer desde coverage, en consola se puede apreciar que algunos test tienen errores del tipo **“Error Got null response for request …”.** Esto no ocurre siempre, pues si se hace desde debug no pasa, incluso también cuando se hace desde coverage hay veces que no ocurre.

**Cobertura**

En esta sección, analizaremos la cobertura de código lograda con nuestros tests. La cobertura de código indica qué porcentaje del código fuente ha sido ejecutado durante las pruebas, ayudando a identificar áreas verificadas y posibles errores no detectados.

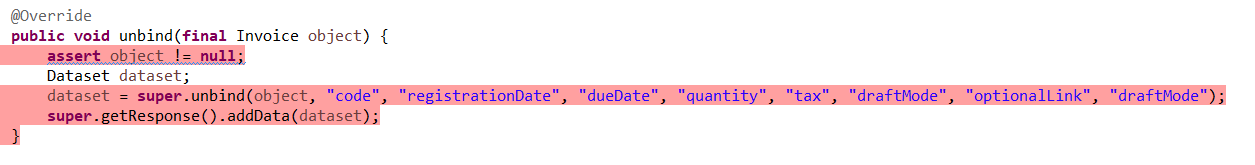
Discutiremos los métodos de medición, presentaremos los resultados obtenidos y evaluaremos la efectividad de nuestros tests. También identificaremos lagunas en la cobertura y propondremos estrategias para mejorarla, asegurando así mayor confiabilidad y robustez del software.

**Cobertura de Invoice:**

****

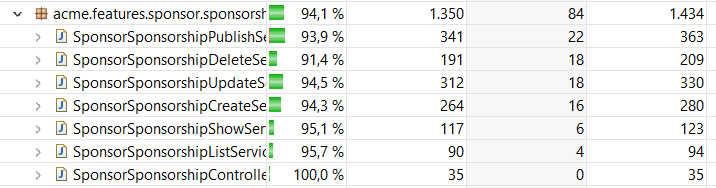
Como se puede apreciar, la cobertura sobre las features de la entidad Invoice es bastante buena, ya que todas las features implementadas tienen una cobertura de más del 90%, salvando el delete.

Esto ocurre porque el delete cuenta con el método unbind, pero este no es nunca usado porque no es necesario para esta funcionalidad, si comentamos este código o lo borramos la cobertura ya superaría el 90%, es por esto que no nos preocupa no llegar al 90% de cobertura.



**Cobertura de sponsorship.**

Con respecto a esta entidad, ocurre una cosa parecida a la comentada anteriormente, ya que la cobertura de todas las features es de mayor que el 90% exceptuando a delete que por el unbind no supera este porcentaje.

****