Universidad de Sevilla  
Escuela Técnica Superior de Ingeniería Informática



**Grado en Ingeniería Informática – Ingeniería del Software**  
**Diseño y Pruebas II**

Curso 2023 – 2024

|  |  |
| --- | --- |
| **Fecha** | **Versión** |
| 27/05/24 | 1.0 |

|  |  |
| --- | --- |
| **Grupo de Prácticas: C1.012** | |
| **Repositorio:** [https://github.com/DP2-C1-012/Acme-SF](mailto:https://github.com/DP2-C1-012/Acme-SF) | |
| **Autores por orden alfabético** | **Correo** |
| Bernal Caunedo, Álvaro Jesús | [alvbercau@alum.us.es](mailto:alvbercau@alum.us.es) |
| Caballero Hernández, Jaime | [jaicabher1@alum.us.es](mailto:jaicabher1@alum.us.es) |
| Casamitjana Benítez, Juan José | [juacasben@alum.us.es](mailto:juacasben@alum.us.es) |
| Herrera Lobo, Nicolás | [nicherlob@alum.us.es](mailto:nicherlob@alum.us.es) |
| Montoya Albitres, Ronald Alexander | [ronmonalb@alum.us.es](mailto:ronmonalb@alum.us.es) |

**Índice de contenido**

[1. Resumen Ejecutivo 2](#_Toc474589782)

[2. Tabla de revisiones 2](#_Toc179714428)

[3. Introducción 3](#_Toc1184956288)

[4. Contenido 3](#_Toc647377246)

[5. Conclusiones 5](#_Toc201131074)

[6. Bibliografía 6](#_Toc1190488549)

# **Resumen Ejecutivo**

Este informe presenta un análisis exhaustivo de las pruebas funcionales realizadas sobre las funcionalidades de facturación y patrocinio de un sistema. El objetivo principal es garantizar la calidad y seguridad del sistema mediante la identificación y corrección de errores potenciales antes de su implementación final. Las pruebas incluyen verificaciones detalladas de creación, eliminación, listado, publicación, visualización y actualización de facturas y patrocinios. Además, se realizaron pruebas de seguridad mediante técnicas de hacking para asegurar que no se puedan manipular los datos por usuarios no autorizados. Los resultados indican que se encontraron y corrigieron varias validaciones faltantes, y no se identificaron errores en las pruebas de seguridad. Este informe subraya la importancia de las pruebas continuas y las evaluaciones periódicas para mantener la funcionalidad y seguridad del sistema, proporcionando así una base sólida para la implementación exitosa de las funcionalidades evaluadas.

# **Tabla de revisiones**

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Revisión** | **Descripción** |
| 27/05/2024 | 1.0 | Versión inicial del documento |
|  |  |  |

# **Introducción**

Este documento presenta una estructura clara y ordenada, con un resumen ejecutivo al inicio seguido de una tabla de revisiones. La introducción establece el contexto, mientras que el contenido aborda los temas principales. Las conclusiones resumen los hallazgos, y la bibliografía respalda la investigación. Esta disposición facilita la comprensión y la navegación del documento.

# **Contenido**

* 1. **Pruebas Funcionales**
     1. **Invoices**
* **Create.safe:** Se comprueban todas las validaciones de creación de facturas al igual que los campos mínimos y máximos (en el caso de esta entidad, la mayoría son costes o campos opcionales). Esta prueba sirvió para encontrar algunas validaciones que faltaban.
* **Delete.safe:** Se comprueba la eliminación de una factura correctamente. En esta prueba no se identificaron errores.
* **List-by-sponsorship.safe:** Se comprueba el listado de facturas para un patrocinio. En esta prueba no se identificaron errores.
* **Publish.safe:** Se comprueban las validaciones de publicación de facturas al igual que los campos y máximos (en el caso de esta entidad, la mayoría son costes o campos opcionales). Esta prueba sirvió para encontrar algunas validaciones que faltaban.
* **Show.safe:** Se comprueba la vista de detalle de una factura. En esta prueba no se identificaron errores.
* **Update.safe:** Se comprueban las validaciones de actualización de facturas al igual que los campos y máximos (en el caso de esta entidad, la mayoría son costes o campos opcionales). Esta prueba sirvió para encontrar algunas validaciones que faltaban.
* **Delete.hack:** Mediante GET hacking, se prueba a eliminar una invoice que no pertenece al sponsor autenticado. En esta prueba no se identificaron errores.
* **List-by-sponsorship.hack:** Mediante GET hacking, se prueba a acceder al listado de facturas iniciando sesión con otro rol. En esta prueba no se identificaron errores.
* **Show.hack:** Mediante GET hacking, se prueba a acceder a la vista de detalle de una factura que pertenece a otro patrocinador.
  + 1. **Sponsorship**

- **Create.safe**: Se comprueban todas las validaciones de creación de patrocinios al igual que los campos mínimos y máximos. Esta prueba sirvió para encontrar algunas validaciones que faltaban.

- **Delete.safe:** Se comprueba la eliminación de un patrocinio correctamente. En esta prueba no se identificaron errores.

- **List-mine.safe:** Se comprueba el listado de patrocinios para un patrocinador. En esta prueba no se identificaron errores.

- **Publish.safe**: Se comprueban las validaciones de publicación de patrocinios al igual que los campos y máximos. Esta prueba sirvió para encontrar algunas validaciones que faltaban.

- **Show.safe**: Se comprueba la vista de detalle de un patrocinio. En esta prueba no se identificaron errores.

- **Update.safe:** Se comprueban las validaciones de actualización de patrocinios al igual que los campos y máximos. Esta prueba sirvió para encontrar algunas validaciones que faltaban.

- **Delete.hack:** Mediante GET hacking, se prueba a eliminar un patrocinio que no pertenece al sponsor autenticado. En esta prueba no se identificaron errores.

- **List-mine.hack:** Mediante GET hacking, se prueba a acceder al listado de patrocinios iniciando sesión con otro rol. En esta prueba no se identificaron errores.

- **Show.hack:** Mediante GET hacking, se prueba a acceder a la vista de detalle de un patrocinio que pertenece a otro patrocinador.

* **Create.hack:** Mediante POST hacking, se prueba a crear un patrocinio para un proyecto no publicado (modificando su id inspeccionando elemento en el navegador)
* **Update.hack:** Mediante POST hacking, se prueba a actualizar un patrocinio con un proyecto no publicado (modificando su id inspeccionando elemento en el navegador).
* **Publish.hack:** Mediante POST hacking, se prueba a actualizar un patrocinio con un proyecto no publicado (modificando su id inspeccionando elemento en el navegador).

Interfaz de usuario gráfica, Aplicación, Tabla, Teams

Descripción generada automáticamente

Figura 1: Cobertura de pruebas

* 1. **Pruebas de Rendimiento**
* **Ordenador 1:**

Gráfico, Gráfico en cascada

Descripción generada automáticamente

Figura 2: Tiempo promedio de las peticiones del pc 1

Cómo podemos ver en la Figura 2, el tiempo promedio de las peticiones es bastante bueno, a excepción de la acción de borrado de facturas que ha tardado considerablemente más.

Interfaz de usuario gráfica, Aplicación, Tabla, Excel

Descripción generada automáticamente

Cuadro 1: Tabla de rendimiento del PC1

En el cuadro 1, se muestran las estadísticas de rendimiento, así como el nivel de confianza. Cabe destacar que los ordenadores que se van a utilizar, ambos tienen escasos recursos, por lo que es normal esa media.

* **Ordenador 3:**

**Gráfico, Gráfico de barras, Gráfico en cascada

Descripción generada automáticamente**

Figura 3: Tiempo promedio de las peticiones del pc2

En la Figura 3 aparece la gráfica de tiempos generados en el segundo ordenador una vez modificado los índices. En comparación a la Figura 2, ha habido una mejora en el tiempo medio de estas peticiones.

Interfaz de usuario gráfica, Aplicación, Tabla, Excel

Descripción generada automáticamente

Cuadro 2: rendimiento de pc2

Por último, en el cuadro 2 aparecen las estadísticas de rendimiento del pc2; cómo podemos ver, el resultado de las modificaciones no ha sido significativo debido a similaridad en los recursos de ambos ordenadores.

* **Comparación:**

**Interfaz de usuario gráfica, Aplicación, Tabla, Excel

Descripción generada automáticamente**

Cuadro 3: Comparativa de rendimiento

En el cuadro 3 se muestran las estadísticas generadas para comparar ambas ejecuciones. Viendo el valor crítico de z para dos colas, se puede concluir en que los cambios no han sido significativos ya que el valor de alfa es 0,05.

# **Conclusi ones**

En conclusión, el proceso de pruebas es esencial para garantizar la calidad y funcionalidad de cualquier sistema. A través del análisis de los distintos componentes y escenarios de uso, se pueden identificar y corregir posibles errores antes de que afecten al usuario final. En este informe de pruebas, se ha presentado un análisis detallado de las pruebas funcionales realizadas sobre las funcionalidades de facturación y patrocinio.

Para las pruebas de facturación, se llevaron a cabo verificaciones exhaustivas en la creación, eliminación, listado, publicación, visualización y actualización de facturas. Se identificaron algunas validaciones que faltaban, lo cual permitió mejorar la robustez del sistema. Además, se realizaron pruebas de seguridad mediante técnicas de hacking para asegurarse de que las facturas no puedan ser manipuladas por usuarios no autorizados, resultando en todas ellas sin errores.

En cuanto a las pruebas de patrocinio, se evaluaron las validaciones durante la creación, eliminación, listado, publicación, visualización y actualización de patrocinios. Al igual que con las facturas, se encontraron algunas validaciones que faltaban y se corrigieron. Las pruebas de seguridad con hacking también confirmaron que los patrocinios no podían ser alterados por usuarios no autorizados, sin identificarse errores en estos casos.

Es importante destacar que la realización de pruebas es un proceso continuo y debe revisarse y ajustarse regularmente para garantizar que el sistema esté funcionando correctamente. Se recomienda realizar evaluaciones periódicas para asegurarse de que el sistema continúe operando según lo esperado y realizar ajustes si es necesario.

En resumen, este informe de pruebas proporciona una guía clara y detallada sobre las verificaciones realizadas, lo cual asegura que las funcionalidades de facturación y patrocinio operen de manera eficiente y segura. Este proceso de pruebas contribuye significativamente a la calidad del sistema y la satisfacción del usuario final.

# **Bibliografía**

Intencionalmente en blanco.