



Escuela Técnica Superior de  
**Ingeniería Informática**

# Testing report Do4

Group C1.020 | Diseño y Pruebas II | 16/05/2024

| Fecha      | Versión | Autor                 |
|------------|---------|-----------------------|
| 16-05-2024 | 1.0     | Gabriel Vacaro Goytia |
|            |         |                       |

**Miembros:**

- Gabriel Vacaro Goytia (gabvacgoy@alum.us.es)

Repositorio de Github: <https://github.com/DP2-C1-020/Acme-SF-Do4>

Contenido

Resumen ejecutivo ..... 3

Introducción ..... 4

Contenido ..... 5

    Pruebas funcionales: ..... 5

    Análisis de desempeño: ..... 8

Bibliografía..... 12

## Resumen ejecutivo

El propósito de este informe es ofrecer una descripción detallada de los diferentes procedimientos seguidos que deben considerarse en el ámbito del testing formal del proyecto Acme-SF, desde la generación de suficientes datos de ejemplo, hasta la meticulosa comprobación de la correcta implementación de los requisitos funcionales y las distintas herramientas estadísticas para comparar los tiempos de ejecución entre peticiones, las diferencias del rendimiento entre dos ordenadores diferentes (antes de la mejora de índices), y por último la comparación de un mismo ordenador (antes y después de la mejora de la base de datos mediante índices).

En resumen, se ha utilizado un enfoque diligente para abordar y solucionar los errores encontrados durante el proceso de testeo con la finalidad de asegurar un producto de alto nivel con el que se satisfagan las expectativas del cliente y se han generado documentos que lo avalan.

## Introducción

En este tipo de informe, se pretende realizar una explicación de como se ha realizado el testing funcional y el análisis de desempeño, con el objetivo de entregar un proyecto de alto nivel con garantías. Para lograrlo, se lleva a cabo una serie de pruebas en relación con los requisitos funcionales 6 y 7.

Este escrito se ha organizado según el documento de anexo proporcionado en enseñanza virtual, en primer lugar, con una portada con las credenciales del autor del reporte, una tabla de versiones en la que se especifican las modificaciones realizadas en este documento clasificadas por número y con sus fechas respectivas; seguidamente un resumen ejecutivo el cual pretende poner en contexto al lector sobre el contenido del documento, una introducción al documento donde se describe el contenido de forma sucinta y se trata la estructura del documento en este último párrafo introductorio.

El trabajo presenta dos apartados clave: uno sobre pruebas funcionales y otro sobre pruebas de rendimiento. En el primero, se detallan los casos de prueba implementados, organizados por característica, con evaluaciones sobre su efectividad para detectar errores. En el segundo, se proporcionan gráficos pertinentes y un intervalo de confianza del 95% para el tiempo de respuesta del proyecto ante las solicitudes funcionales y la comparación pertinente. Estos capítulos constituyen una evaluación exhaustiva de la calidad y eficiencia del proyecto en desarrollo, por último, una conclusión del analista y la bibliografía utilizada.

## Contenido

### Pruebas funcionales:

Para cada característica se han realizado tanto pruebas positivas y negativas (X.safe) como intentos de hacking (Y.hack), para la realización de estas pruebas, se ha seguido las recomendaciones dictadas en el documento “L04-S01 – Formal testing”:

Para no caer en redundancias, listaré las pruebas realizadas según el tipo de operación, pues esto será lo que determinará el modo de actuar de la prueba:

### OPERACIONES "LIST-MINE"

#### *Pruebas Seguras:*

- Listar todos los proyectos o historias de usuario de varios gerentes.

#### *Pruebas de Hacking:*

- Intentar acceder a la URL con un rol incorrecto.

### OPERACIONES "SHOW"

#### *Pruebas Seguras:*

- Seleccionar varios proyectos o historias de usuario de una lista para ver sus detalles.

#### *Pruebas de Hacking:*

- Intentar acceder a los detalles de un proyecto o historia de usuario con un rol incorrecto o con un gerente diferente.
- Intentar acceder a un proyecto o historia de usuario inexistente con un identificador inválido.

### OPERACIONES "CREATE"

#### *Pruebas Seguras:*

- Intentar crear un nuevo proyecto o historia de usuario.
- Verificar que el sistema rechace datos inválidos para cada atributo.
- Verificar que el sistema acepte datos válidos para cada atributo.

#### *Pruebas de Hacking:*

- No se encontraron formas de hacking.

## OPERACIONES "UPDATE"

### *Pruebas Seguras:*

- Actualizar un proyecto o historia de usuario existente.
- Verificar que el sistema rechace datos inválidos para cada atributo.
- Verificar que el sistema acepte datos válidos para cada atributo.

### *Pruebas de Hacking:*

- Probar URLs de forma maliciosa para ver si se encuentran vulnerabilidades (GET hacking).
- Cambiar datos mediante la herramienta inspeccionar del navegador, y posteriormente intentar actualizarlos mediante GET hacking.

## OPERACIONES "DELETE"

### *Pruebas Seguras:*

- Intentar eliminar proyectos o historias de usuario, considerando restricciones como elementos dependientes o estados de publicación.
- Eliminar correctamente un proyecto o historia de usuario cuando es posible.

### *Pruebas de Hacking:*

- Intentar acceder al borrado de un proyecto sin tener los permisos de este mediante URLs

## OPERACIONES "PUBLISH"

### *Pruebas Seguras:*

- Intentar publicar proyectos o historias de usuario, considerando condiciones previas necesarias para la publicación.
- Verificar el proceso de actualización de atributos durante la publicación.
- Intentar publicar con errores fatales.

### *Pruebas de Hacking:*

- No se encontraron formas de hacking.

## OPERACIONES EN LA TABLA INTERMEDIA "PROJECTUSERSTORY"

### *Pruebas Seguras:*

- Vincular y desvincular proyectos a historias de usuario.
- Verificar que se rechacen asociaciones o desvinculaciones inválidas.
- Asegurar que las asociaciones y desvinculaciones se manejen correctamente según el estado de publicación.

*Pruebas de Hacking:*

- Intentar acceder a las páginas de vinculación y desvinculación con un rol incorrecto o con un gerente diferente.

Por último, véase aquí las pruebas implementadas y una categorización sobre su efectividad en esta última fase del proyecto y el coverage de cada feature.

| Ruta de la feature              | X.safe   | Y.hack                                | Efectividad  | Coverage |
|---------------------------------|--|---------------------------------------|--|----------|
| /any/...                        | List-project<br>Show-project                               | Show-project                          | Baja, No se encontraron fallos ni vulnerabilidades.  | 94.4%    |
| /authenticated/manager/...      | Créate<br>Show<br>Update                                   | Show<br>Update                        | Alta, en este apartado se encuentra el MIR, en concreto al actualizar un manager.  | 88.8%    |
| /manager/Dashboard/...          | Show   | Show                                  | Baja, No se encontraron fallos ni vulnerabilidades.  | 98.4%    |
| /manager/Project/...            | Créate<br>Delete<br>List-mine<br>Show<br>Update<br>Publish | Delete<br>List-mine<br>Show<br>Update | Muy alta, se depuraron algunas líneas de impresión por pantalla pensadas para el debug que no deberían estar y un bug que, bajo ciertas condiciones provocaba que al borrar un proyecto no se borrarán algunas historias de usuario. | 91.4%    |
| /manager/Project-user-story/... | Link<br>Unlink   | Link<br>Unlink                        | Alta, encontrado bug que bajo ciertos datos de prueba las relaciones en las que el proyecto era nulo ocasionaban un panic.   | 89.7%    |
| /manager/User-stories/...       | Créate<br>Delete<br>List-mine<br>Show<br>Update<br>Publish | Delete<br>List-mine<br>Show<br>Update | Media, bug menor encontrado que hacía comprobaciones redundantes al borrar una historia de usuario bajo ciertas condiciones.   | 78,6%    |

## Análisis de desempeño:

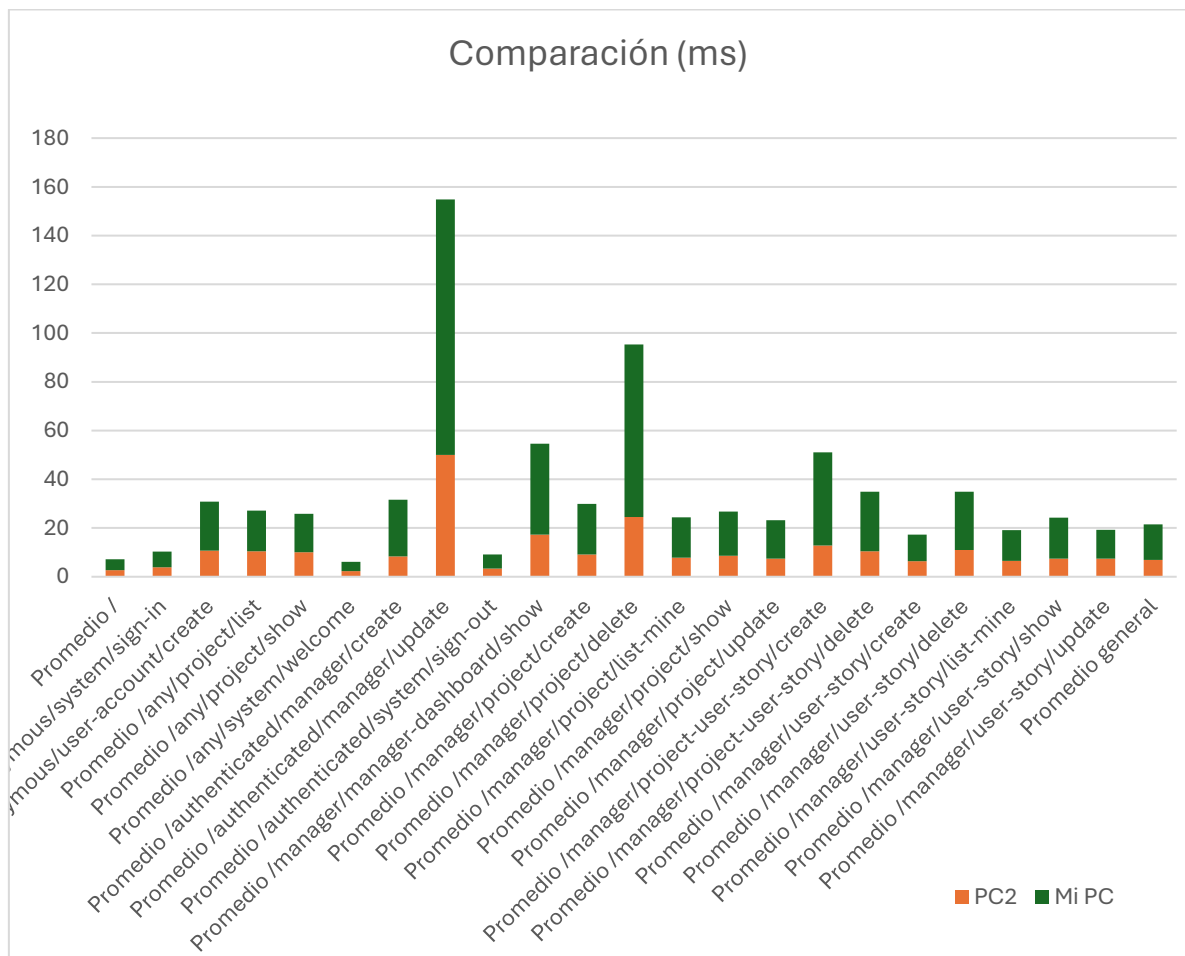
En este apartado veremos las estadísticas propias de la aplicación:

### 1) Sin optimizar índices en dos ordenadores distintos:

| STATISTICS-PC2              |             |              |             |             | time-pc2  | time-miPc | STATISTICS-MYPC           |             |              |             |             |
|-----------------------------|-------------|--------------|-------------|-------------|-----------|-----------|---------------------------|-------------|--------------|-------------|-------------|
| Media                       | 7,032846065 | Interval(ms) | 6,205195166 | 7,860496964 | 4,3535    | 7,5932    | Media                     | 15,05672832 | Interval(ms) | 12,81553318 | 17,29792346 |
| Error típico                | 0,421209404 | Interval(s)  | 0,006205195 | 0,007860497 | 3,052101  | 5,1372    | Error típico              | 1,148622545 | Interval(s)  | 0,012815533 | 0,017297923 |
| Mediana                     | 5,677401    |              |             |             | 3,232201  | 5,7233    | Mediana                   | 9,55035     |              |             |             |
| Moda                        | 5,6025      |              |             |             | 2,650201  | 4,5814    | Moda                      | 3,7192      |              |             |             |
| Desviación estándar         | 9,218617915 |              |             |             | 4,6674    | 5,0114    | Desviación estándar       | 25,09369599 |              |             |             |
| Varianza de la muestra      | 84,98291626 |              |             |             | 2,9709    | 4,2208    | Varianza de la muestra    | 629,6935786 |              |             |             |
| Curtosis                    | 150,9159598 |              |             |             | 3,2419    | 5,1148    | Curtosis                  | 60,79906946 |              |             |             |
| Coefficiente de asimetría   | 10,23075059 |              |             |             | 3,323801  | 6,1338    | Coefficiente de asimetría | 6,815479595 |              |             |             |
| 1 Rango                     | 155,122699  |              |             |             | 18,823899 | 6,5744    | Rango                     | 291,9664    |              |             |             |
| 2 Mínimo                    | 1,6883      |              |             |             | 3,345701  | 4,2615    | Mínimo                    | 2,6699      |              |             |             |
| 3 Máximo                    | 156,810999  |              |             |             | 2,3352    | 3,9969    | Máximo                    | 294,6363    |              |             |             |
| 4 Suma                      | 3368,733265 |              |             |             | 3,1075    | 4,1644    | Suma                      | 7287,456506 |              |             |             |
| 5 Cuenta                    | 479         |              |             |             | 3,4948    | 7,0456    | Cuenta                    | 484         |              |             |             |
| 6 Nivel de confianza(95,0%) | 0,827650899 |              |             |             | 3,068899  | 3,9241    | Nivel de confianza(95,0%) | 2,241195142 |              |             |             |

Ilustración 1 Tabla comparativa entre el pc2 y mi pc

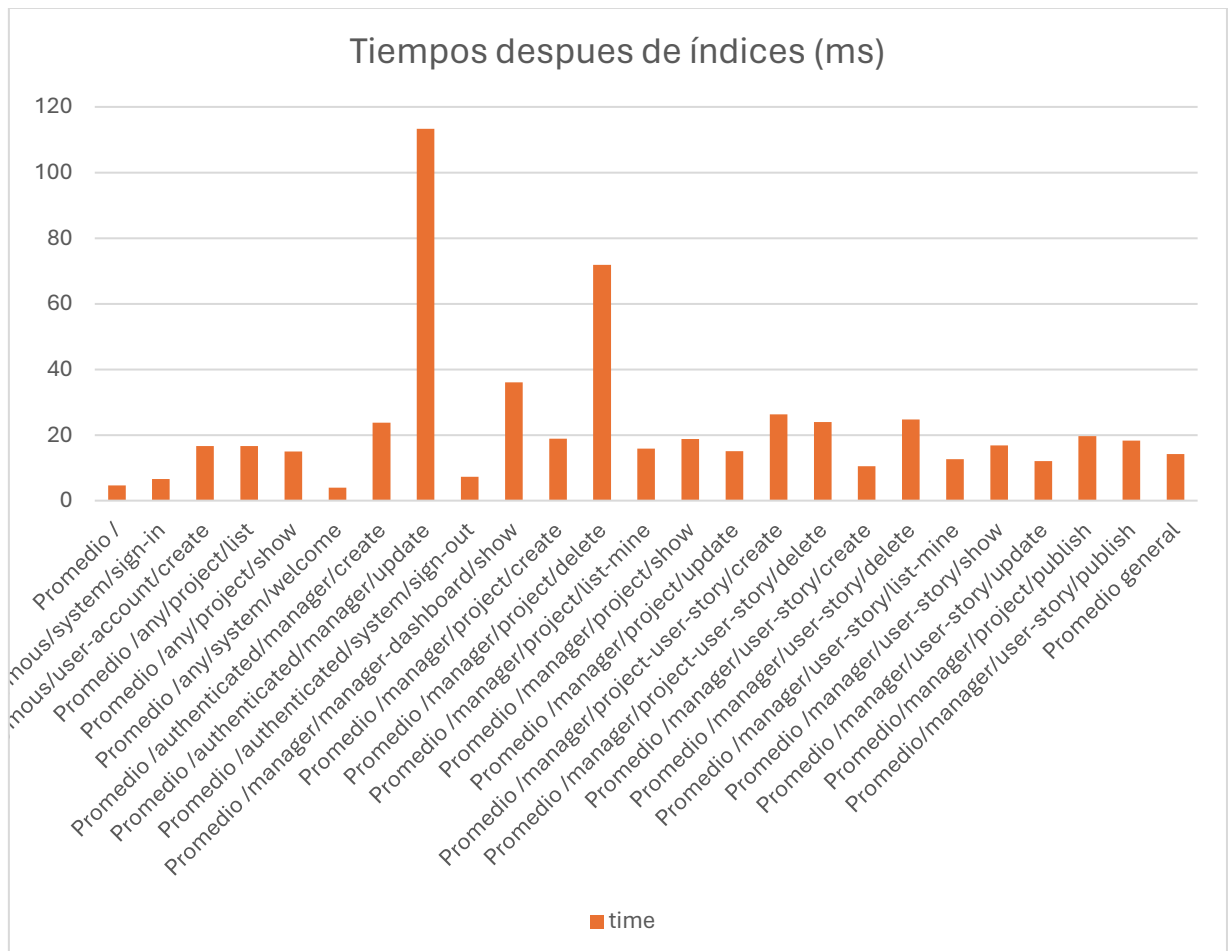
En este caso podemos observar que el pc2 (de mi compañero Guillermo), tiene unas prestaciones mucho mayores a las de mi pc, siendo mucho más constante a la hora de servir las peticiones y superándolo por un “speed up” de 2,1409 en la media, además de tener un intervalo de confianza del 95% mucho más bajo, aunque ambos sean tiempos muy decentes. (los test publish tanto de Project como de user story solo están en la gráfica después de implementar índices)





Como podemos observar, su ordenador claramente tiene unas prestaciones mayores, esto se puede observar sobre todo en el MIR, donde el peor caso se ve aún más acentuado debido.

- 2) Tras realizar la modificación en base de datos para agilizar búsquedas mediante los índices, obtenemos este nuevo resultado:



Podemos apreciar una clara mejora en los métodos que más tiempo consumían debido a su necesidad de acceder a la base de datos, lo cual gracias a los índices se ha mitigado en gran medida.

En las estadísticas, también se ve reflejado en una disminución tanto de la cota superior como de la inferior del intervalo de confianza:

| ESTADÍSTICAS MI PC DESPUES DE INDICES |             |              |           |           |
|---------------------------------------|-------------|--------------|-----------|-----------|
| Media                                 | 14,61002707 | interval(ms) | 12,470293 | 16,749761 |
| Error típico                          | 1,088985392 | interval(s)  | 0,0124703 | 0,0167498 |
| Mediana                               | 9,3568      |              |           |           |
| Moda                                  | 4,1188      |              |           |           |
| Desviación estándar                   | 23,95767862 |              |           |           |
| Varianza de la muestra                | 573,9703649 |              |           |           |
| Curtosis                              | 78,62945671 |              |           |           |
| Coefficiente de asimetría             | 7,464295193 |              |           |           |
| Rango                                 | 327,3614    |              |           |           |
| Mínimo                                | 2,5326      |              |           |           |
| Máximo                                | 329,894     |              |           |           |
| Suma                                  | 7071,2531   |              |           |           |
| Cuenta                                | 484         |              |           |           |
| Nivel de confianza(95,0%)             | 2,139733937 |              |           |           |

Por último, en el contraste de hipótesis, nos damos cuenta de que el p-value (0.7435), se sitúa entre 0.05 y 1, por lo tanto, concluyo que no han sido cambios que se hayan traducido en una mejora de gran significancia, aunque podamos ver que la media es débilmente menor.

| Prueba z para medias de dos muestras |             |           |
|--------------------------------------|-------------|-----------|
|                                      |             |           |
|                                      | 294,6363    | 329,894   |
| Media                                | 14,0425449  | 13,514336 |
| Varianza (conocida)                  | 631,48215   | 570,18472 |
| Observaciones                        | 461         | 461       |
| Diferencia hipotética de las medias  | 0           |           |
| z                                    | 0,327162959 |           |
| P(Z<=z) una cola                     | 0,371772317 |           |
| Valor crítico de z (una cola)        | 1,644853627 |           |
| Valor crítico de z (dos colas)       | 0,743544635 |           |
| Valor crítico de z (dos colas)       | 1,959963985 |           |

## Conclusiones

Sin esta fase de testing formal, no hubiera sido capaz de aseverar con una rigurosidad de nivel profesional que la calidad de mi producto era alta. Esta etapa ha sido esencial no solo para identificar y corregir errores, sino también para asegurarme de que cada aspecto de la aplicación funciona según lo esperado bajo diversas condiciones. Este proceso meticuloso me ha permitido detectar fallos críticos que podrían haber pasado desapercibidos en una revisión más superficial, que, de no haber sido corregidos, podrían haberse traducido en problemas graves y potenciales fallos durante el uso por parte del cliente, afectando su experiencia y la reputación del producto.

Gracias al testing formal, he podido asegurarme de que el producto final es robusto, fiable y capaz de cumplir con las expectativas y necesidades del usuario. Este nivel de escrutinio no solo mejora la calidad del producto, sino que también genera una mayor confianza entre los usuarios y clientes, quienes pueden estar seguros de que están adquiriendo un producto bien desarrollado y probado rigurosamente.

Finalmente, el uso de comprobaciones en distintos ordenadores, así como el elemento estadístico utilizado sobre los datos del producto, como el Z-Test y el intervalo de confianza, han sido fundamentales para validar el rendimiento y la estabilidad de la aplicación. Estas herramientas estadísticas nos han permitido evaluar de manera objetiva si los cambios realizados han tenido un impacto significativo. El Z-Test, en particular, ha proporcionado una forma precisa de comparar los tiempos de ejecución antes y después de las modificaciones, mientras que el intervalo de confianza nos ha ofrecido una medida de la precisión de nuestras estimaciones. Estas metodologías han asegurado que los resultados obtenidos no son producto del azar, sino reflejan mejoras o consistencias reales en el rendimiento del producto.

En resumen, el testing formal no es solo una fase más del desarrollo, sino una piedra angular que asegura la calidad del producto protege la satisfacción del cliente y fortalece la confianza en los productos que desarrollo.

## Bibliografía

Web de la universidad de Sevilla - <https://ev.us.es>