

INFORME DE PRUEBAS

ESTUDIANTE #1
Ruiz Gutiérrez, Álvaro
(alvruigut@alum.us.es)

16/02/2024

Grupo : C1.030
Repositorio:
<https://github.com/yesgarfue/Acme-SF-Do4.git>

ÍNDICE

1. <i>Historial de versiones</i>	3
2. <i>Introducción</i>	4
3. <i>Contenidos</i>	5
4. <i>Testing Funcional</i>	6
5. <i>Testing Rendimiento</i>	8
6. <i>Conclusión</i>	12
7. <i>Bibliografía</i>	14

HISTORIAL DE VERSIONES

Versión	Fecha	Descripción
1.0	10/05/2024	Creación del documento
1.1	17/05/2024	Entregable D04

INTRODUCCIÓN

En el siguiente documento, se detallan los dos test suites propuestas para el estudiante 1, y se realizan unas pruebas de rendimiento para verificar que la aplicación funciona de forma eficiente y sin errores.

Se pretende realizar un análisis de las pruebas realizadas por el estudiante 1 a algunos requisitos del proyecto Acme-SF, que nos ayudará a comprobar que el proyecto se encuentra en la medida de lo posible libre de errores y vulnerabilidades que puedan poner en peligro la integridad del sistema.

CONTENIDOS

DO4

- TAREA 06:

“Operations by managers on projects:

- List the projects that they have created.
- Show the details of their projects.
- Create, update, or delete their projects. Projects can be updated or deleted as long as they have not been published. For a project to be published, it must have at least one user story, and all its user stories must have been published. Moreover, it must not have any fatal errors...”

Testing Funcional

Se han ejecutados pruebas positivas para comprobar que el sistema funciona debidamente en los servicios de listar, mostrar, crear, actualizar, publicar y eliminar. Tratando de aumentar el porcentaje de coverage en el código al máximo, tratando de evitar bugs en el código y líneas innecesarias.

Para los test de pruebas negativos, se han efectuado en el mismo archivo con extensión .safe de los servicios crear, actualizar y publicar, en la carpeta de test/resources.

En los tests de hacking, se ha tratado de realizar acciones en los servicios del usuario manager1 para evitar un uso malicioso de la aplicación.

Projects

- ManagerProjectList:
 - Comprueba que los datos del listado que se muestra concuerdan con los esperados.
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda acceder a listar los proyectos del usuario manager1.
- ManagerProjectShow
 - Comprueba que los datos del listado se corresponden con los datos que se muestran en la vista de detalles de cada proyecto.
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión pueda acceder a mostrar los datos de los proyectos del usuario manager1, incluyendo un usuario con el mismo rol de manager.
- ManagerProjectCreate
 - Crea un proyecto con datos válidos, probando algunos casos límite para comprobar que la aplicación responde bien ante ellos. Luego, se comprueba que se ha creado correctamente. Además, intenta crear un proyecto, pero poniendo algún campo que no cumpla las restricciones y esperando un error.
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda crear proyectos del usuario manager1.
- ManagerProjectUpdate
 - Actualiza los datos de un proyecto con valores que sean válidos, intentando probar los límites de cada campo.
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda modificar proyectos del usuario manager1.
- ManagerProjectDelete
 - Elimina un proyecto que cumpla las restricciones impuestas para que esta acción ocurra (esto es, que dicho proyecto, no se encuentre en modo borrador).
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda eliminar proyectos del usuario manager1.
- ManagerProjectPublish
 - Publica un proyecto que cumpla con las restricciones impuestas para que pueda publicarse, que son que se encuentre en borrador. En el caso negativo, trata de publicar un proyecto con fatal errors, esperando un error. Además, trata de publicar un proyecto efectuando cambios erróneos en el formulario para comprobar el correcto funcionamiento de la aplicación.

- Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda publicar proyectos del usuario manager1.

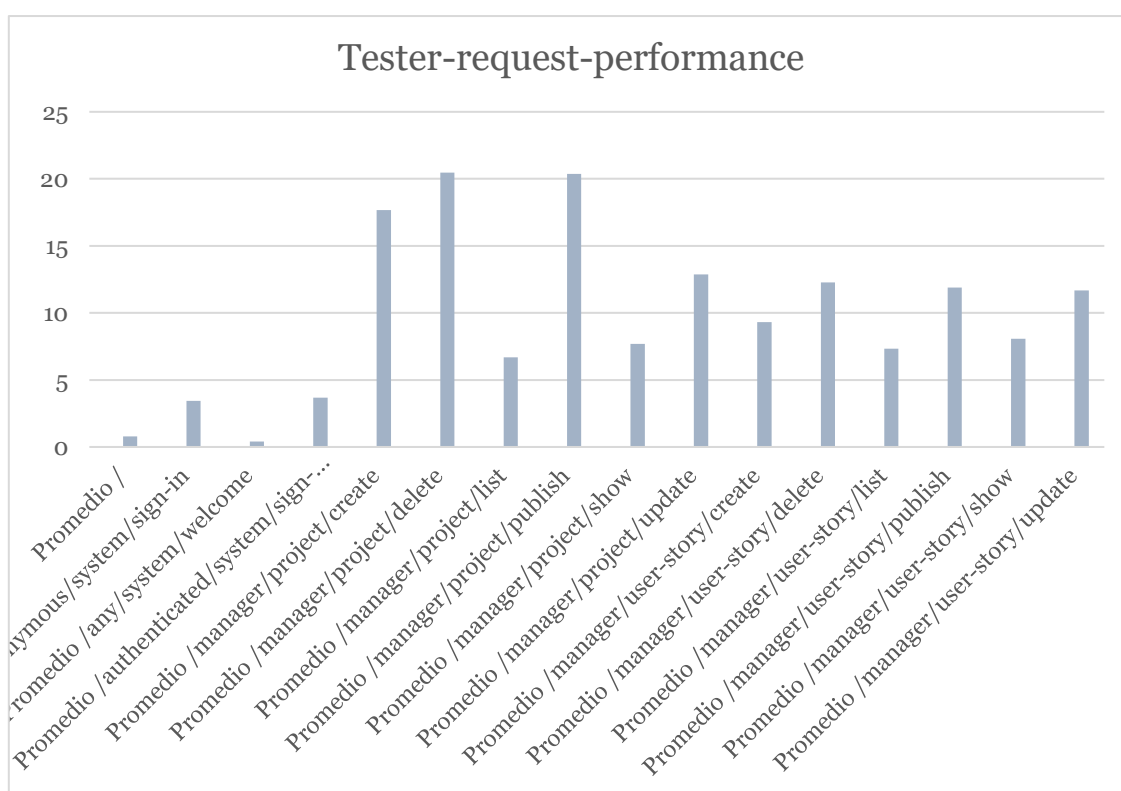
UserStories

- ManagerUserStoryList
 - Comprueba que los datos del listado que se muestra concuerdan con los esperados.
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda acceder a listar las historias de usuario del usuario manager1.
- ManagerUserStoryShow
 - Comprueba que los datos del listado se corresponden con los datos que se muestran en la vista de detalles de cada historia de usuario.
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda acceder a mostrar los datos de las historias de usuario del usuario manager1, incluyendo un usuario con el mismo rol de manager.
- ManagerUserStoryCreate
 - Crea un proyecto con datos válidos, probando algunos casos límite para comprobar que la aplicación responde bien ante ellos. Luego, se comprueba que se ha creado correctamente. Además, intenta crear una historia de usuario, pero poniendo algún campo que no cumpla las restricciones y esperando un error.
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda crear historias de usuario del usuario manager1.
- ManagerUserStoryUpdate
 - Actualiza los datos de una historia de usuario, con valores que sean válidos, intentando probar los límites de cada campo.
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda modificar historias de usuario del usuario manager1.
- ManagerUserStoryDelete
 - Elimina una historia de usuario que cumpla las restricciones impuestas para que esta acción ocurra (no se encuentre en modo borrador).
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda eliminar historias de usuario del usuario manager1.
- ManagerUserStoryPublish
 - Publica una historia de usuario, que cumpla con las restricciones impuestas para que pueda publicarse, que son que se encuentre en borrador. Además, trata de publicar una historia de usuario, efectuando cambios erróneos en el formulario para comprobar el correcto funcionamiento de la aplicación.
 - Hacking: Comprueba que ningún usuario o persona sin iniciar sesión, pueda publicar historias de usuario del usuario manager1.

TESTING DE RENDIMIENTO

Se han realizado a continuación, pruebas de rendimiento en mi PC personal. Para ellos hemos medido el tiempo de ejecución de todos los test, usando algunas herramientas de análisis de datos para algunos ficheros CSV en Excel, con los logs generados en eclipse. Además, se ha usado la aplicación Visualvm para hacer más pruebas de rendimiento, y medido el tiempo de ejecución de los tests con el monitor de rendimiento de mi PC.

Análisis de datos de rendimiento extraídos de los logs:



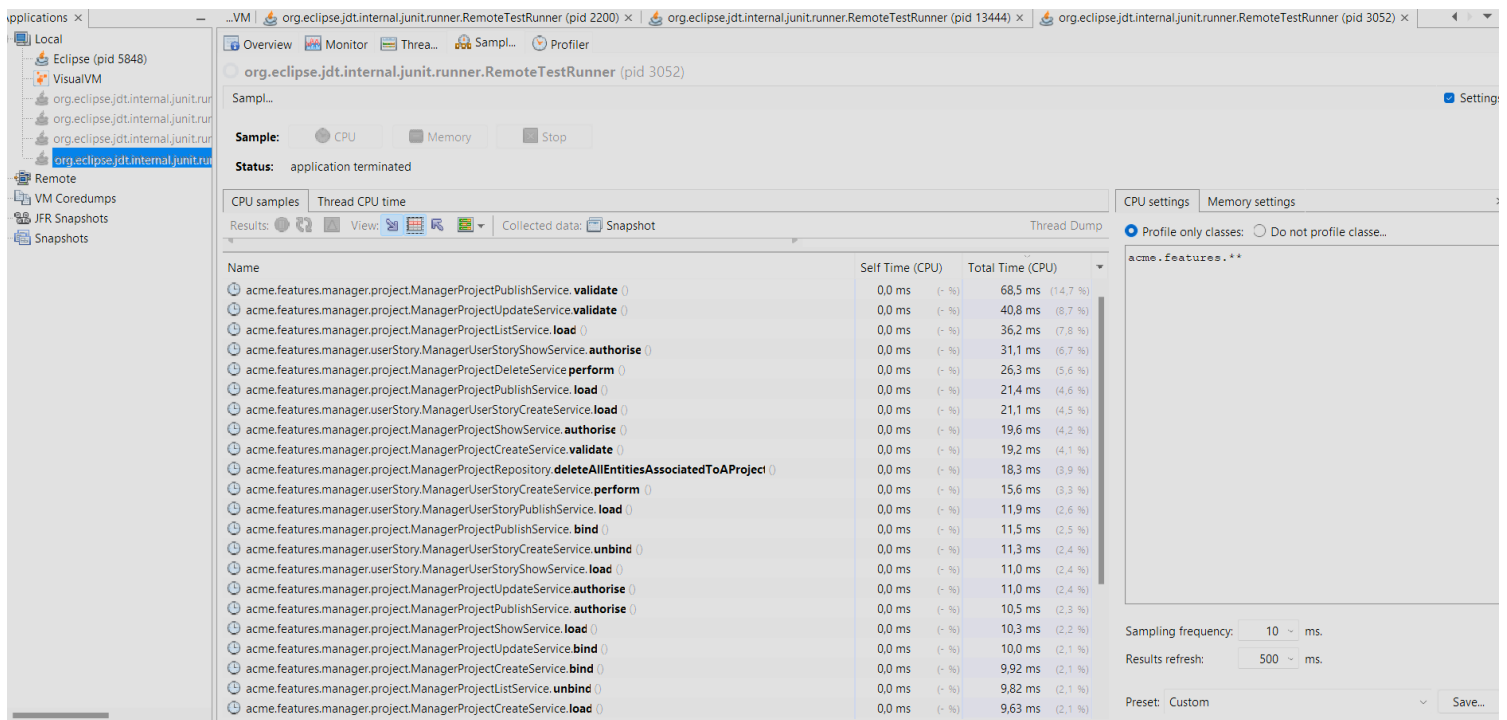
Análisis de tiempo de request performance	
Media	4,84212467
Error típico	0,31313086
Mediana	3,12965
Moda	0,3007
Desviación estándar	6,64250857
Varianza de la muestra	44,1229202
Curtosis	12,3059823
Coeficiente de asimetría	2,87388479
Rango	54,3281
Mínimo	0,2339
Máximo	54,562

Suma	2178,9561
Cuenta	450
Nivel de confianza(95,0%)	0,61538401

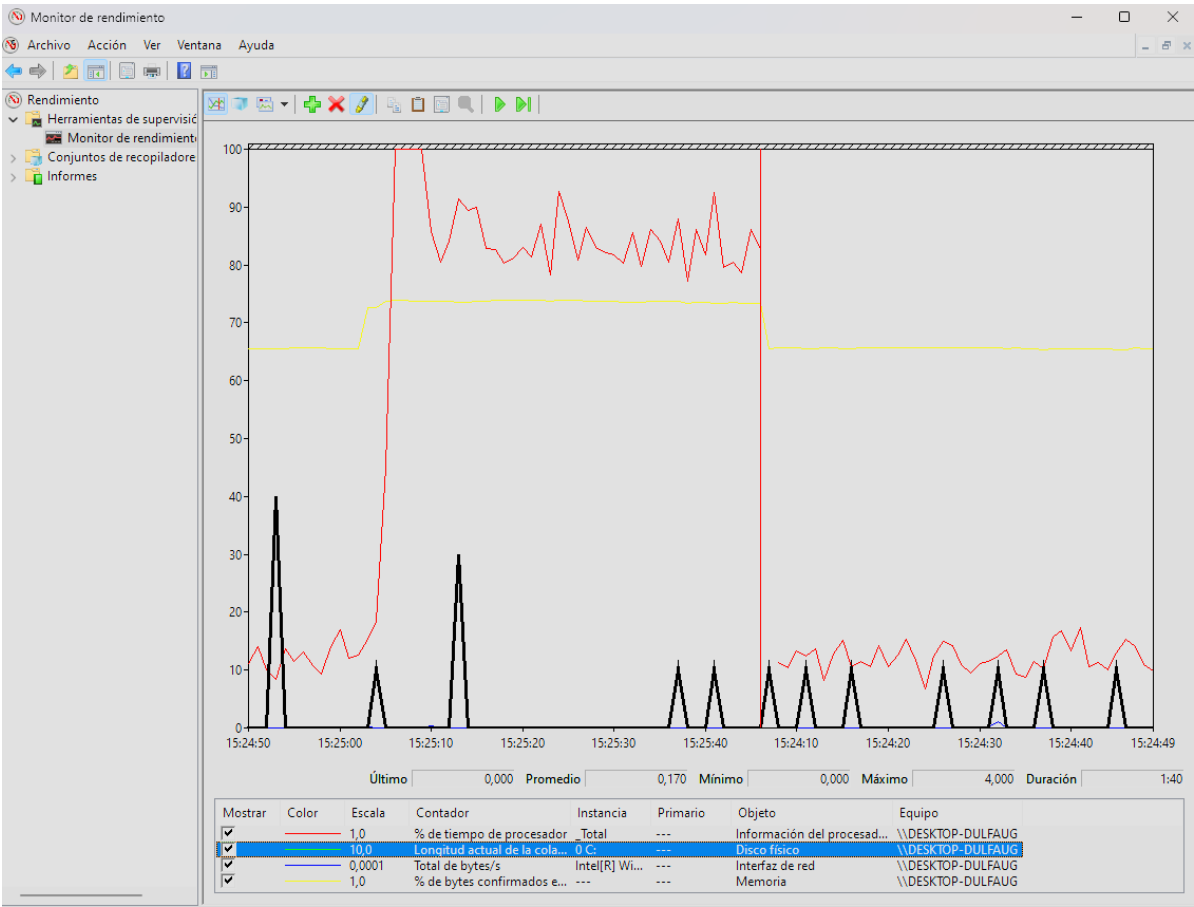
Interval(ms)	4,226740659	5,45750867
Interval(s)	0,004226741	0,00545751

request-path	time
Promedio /	0,797806154
Promedio /anonymous/system/sign-in	3,44985122
Promedio /any/system/welcome	0,408919811
Promedio /authenticated/system/sign-out	3,677326829
Promedio /manager/project/create	17,68273333
Promedio /manager/project/delete	20,46754
Promedio /manager/project/list	6,695633333
Promedio /manager/project/publish	20,37906667
Promedio /manager/project/show	7,695766667
Promedio /manager/project/update	12,87789091
Promedio /manager/user-story/create	9,304106667
Promedio /manager/user-story/delete	12,273775
Promedio /manager/user-story/list	7,321956
Promedio /manager/user-story/publish	11,88764286
Promedio /manager/user-story/show	8,080328571
Promedio /manager/user-story/update	11,68305
Promedio general	4,842124667

Análisis de software mediante VisualVM:



Análisis de hardware vía monitor de rendimiento:



CONCLUSIÓN

Después de un análisis exhaustivo de la aplicación, podemos concluir que se encuentra, en la medida de lo posible, libre de vulnerabilidades y de errores que puedan afectar a la experiencia de usuario, mediante pruebas completas de los requisitos obligatorios. Además, se han hecho varias pruebas de rendimiento para comprobar que la aplicación es suficientemente eficiente para cumplir las restricciones impuestas por el cliente.

BIBLIOGRAFÍA

No procede.