|  |
| --- |
| **INFORME DE TESTEO** |



Grupo: C1.02.14

Repositorio de GitHub: https://github.com/tcabgom/Acme-L3-D04.git

Fecha: 05/25/2023

|  |  |
| --- | --- |
| Autores | Emails |
| Arroyo Mantero, Ignacio | [ignarrman@alum.us.es](mailto:ignarrman@alum.us.es) |

Contenido

[Resumen 3](#_Toc127465407)

[Versión 3](#_Toc127465408)

[Introducción 4](#_Toc127465409)

[Contenido 4](#_Toc127465410)

[Conclusiones 5](#_Toc127465411)

[Bibliografía 5](#_Toc127465412)

|  |
| --- |
| Resumen |

Este documento contiene datos y estadísticas relativos a los resultados de realizar testing formal sobre las funcionalidades de la aplicación.

|  |
| --- |
| Versión |

|  |  |  |
| --- | --- | --- |
| Número de revisión | Fecha | Descripción |
| 1.0 | 25/05/2023 | Creación del documento |

|  |
| --- |
| Introducción |

Este documento contiene una lista de las tareas con su título, descripción, estimación de tiempo y tiempo real que se necesitó para realizar la tarea. Además, contendrá una tabla con el presupuesto de los salarios de este sprint.

|  |
| --- |
| Contenido |

Pruebas Funcionales sobre Audit:

* **AuditorAuditListTest:** Comprueba que la aplicación lista todas las auditorías del auditor y bloquea los intentos de hackeo sobre el listado.
* **AuditorAuditShowTest:** Comprueba que enseña correctamente todos los datos de una auditoría del auditor y rechaza los intentos de hackeo.
* **AuditorAuditCreateTest:** Comprueba que la aplicación guarda correctamente nuevas auditorías y comprueba que no se pueda añadir auditorías con datos erroneos y rechaza intentos de hackeo.
* **AuditorAuditDeleteTest:** Comprueba que las auditorías se pueden borrar correctamente y rechaza intentos de hackeo.
* **AuditorAuditUpdateTest:** Comprueba que las auditoría se pueden actualizar y comprueba que no se puedan actualizar con datos erróneos y rechaza intentos de bloqueo.
* **AuditorAuditPublishTest:** Comprueba que las auditorías se pueden publicar correctamente y rechaza intentos de hackeo.

Pruebas Funcionales sobre AuditingRecords:

* **AuditorAuditingRecordsListTest:** Comprueba que la aplicación lista todos los registros de auditoría de una auditoría del auditor y bloquea los intentos de hackeo sobre el listado.
* **AuditorAuditingRecordsShowTest:** Comprueba que enseña correctamente todos los datos de un registro de auditoría del auditor y rechaza los intentos de hackeo.
* **AuditorAuditingRecordsCreateTest:** Comprueba que la aplicación guarda correctamente nuevos registros de auditorías en una auditoría y comprueba que no se pueda añadir auditorías con datos erroneos y rechaza intentos de hackeo.
* **AuditorAuditingRecordsDeleteTest:** Comprueba que los registros de auditoría se pueden borrar correctamente y rechaza intentos de hackeo.
* **AuditorAuditingRecordsUpdateTest:** Comprueba que los registros de auditoría se pueden actualizar correctamente y comprueba que no se puedan actualizar con datos erróneos o sin confirmación y rechaza intentos de bloqueo.
* **AuditorAuditingRecordsPublishTest:** Comprueba que los registros de auditorías se pueden publicar correctamente y comprueba el caso de que no se use confirmación y rechaza intentos de hackeo.

Rendimiento de las pruebas de Auditoría:

Gráfico

Descripción generada automáticamente Gráfico

Descripción generada automáticamente

En estas gráficas podemos observar que las funcionalidades que más tiempo consumen son las de listado y creación y podemos ver que no hay una diferencia significativa entre los dos ordenadores.

Rendimiento de las pruebas de Auditoría por tipo de método:

Gráfico

Descripción generada automáticamente

Gráfico, Gráfico en cascada

Descripción generada automáticamente

En estas gráficas podemos observar que el método que más consumen es claramente el hacking y una vez más la diferencia entre ambos ordenadores es apenas notable.

Comparación de rendimiento de las pruebas de Auditoría en dos ordenadores:

* Como podemos observar en la tabla los valores de la tabla de Z-test extrañamente parecen coincidir pero como podemos onservar en la media el último dígito no coincide. Esto se debe a que las mejoras sobre las especificaciones no han sido significativas de ninguna forma.

|  |  |  |
| --- | --- | --- |
|  | *Ordenador1* | *Ordenador2* |
| Media | 9,056768559 | 9,05676856 |
| Varianza (conocida) | 77,7867046 | 77,7867046 |
| Observaciones | 458 | 458 |
| Diferencia hipotética de las medias | 0 |  |
| z | 0 |  |
| P(Z<=z) una cola | 0,5 |  |
| Valor crítico de z (una cola) | 1,644853627 |  |
| Valor crítico de z (dos colas) | 1 |  |
| Valor crítico de z (dos colas) | 1,959963985 |  |

Rendimiento de las pruebas de Registros de Auditoría por funcionalidad:

Gráfico, Gráfico de barras

Descripción generada automáticamente Gráfico, Gráfico de barras

Descripción generada automáticamente

En estas gráficas podemos observar que las funcionalidades que más tiempo consumen son las de listado, creación y actualización y podemos ver que no hay una diferencia significativa entre los dos ordenadores.

Rendimiento de las pruebas de Registros de Auditoría por tipo de método:

Gráfico, Gráfico de barras, Gráfico en cascada

Descripción generada automáticamente

Gráfico, Gráfico de barras, Gráfico en cascada

Descripción generada automáticamente

En estas gráficas podemos observar que el método que más consumen es claramente el hacking y una vez más la diferencia entre ambos ordenadores es apenas notable.

Comparación de rendimiento de las pruebas de Registros de Auditoría en dos ordenadores:

* Como podemos observar en la tabla los valores de la tabla de Z-testcoinciden de nuevo. Esto se debe unavez más a que las mejoras sobre las especificaciones no han sido significativas de ninguna forma.

|  |  |  |
| --- | --- | --- |
| Prueba z para medias de dos muestras | | |
|  |  |  |
|  | *time* | *time* |
| Media | 7,98563734 | 7,98563734 |
| Varianza (conocida) | 43,8702969 | 43,8702969 |
| Observaciones | 557 | 557 |
| Diferencia hipotética de las medias | 0 |  |
| z | 0 |  |
| P(Z<=z) una cola | 0,5 |  |
| Valor crítico de z (una cola) | 1,64485363 |  |
| Valor crítico de z (dos colas) | 1 |  |
| Valor crítico de z (dos colas) | 1,95996398 |  |

|  |
| --- |
| Conclusiones |

Según las pruebas podemos asegurar que la aplicación funciona correctamente.

|  |
| --- |
| Bibliografía |

Intencionalmente en blanco.