|  |
| --- |
| **Test**  **Report** |
| **Miembros:**  **1- Juan Antonio Moreno Moguel (juamormog@alum.us.es)** |
| **26-04-2024**  **Grupo: C2.003 Repositorio:**  **https://github.com/IsmaelRuizJurado/Acme- SF-D04** |

**1**



[Resumen Ejecutivo 3](#_bookmark0)

[Historial de Versiones 4](#_bookmark1)

[Introducción 5](#_bookmark2)

[Contenidos 6](#_bookmark3)

[Conclusiones 6](#_bookmark3)

[Bibliografía 10](#_bookmark4)

En este reporte se verá reflejado el resultado de los tests creados para las funciones pedidas durante el proyecto

|  |  |  |
| --- | --- | --- |
| Versión | Fecha | Descripción de los Cambios |
| 1.0 | 25/05/2024 | Creación y revisión del documento |
| 2.0 | 26/06/2024 | Actualización para segunda evaluación |

Este documento tratará la información dada tras la realización de las pruebas hechas a las funciones añadidas a la aplicación, este documento se estructurará en dos partes.

La primera, tratara las pruebas funcionales, proporcionando una lista de casos de prueba usados y descritos.

La segunda parte se enfocará en las pruebas de rendimiento en la que se mostrarán gráficos que darán un intervalo de confianza del 95% para el tiempo de respuesta del proyecto. Además, se incluirá un contraste de hipótesis para determinar con los datos de otro compañero para determinar que ordenador es más potente.

## Test de funcionalidad

A continuación, detallaremos las pruebas que se han realizado para testear el funcionamiento del software, se explicara que se ha hecho, los resultados esperados, el real, y que bugs encontrado en el caso de que sea así. En el caso de que haya habido bugs, se han arreglado y vuelto a realizar el test.

Descripcion de test: Se detalla la prueba junto a los pasos llevados a cabo en ella Resultado esperado: Comportamiento que se esperaba obtener de la prueba Resultado real: Resultado obtenido de ella

Bugs encontrados: Bugs que se hayan encontrado y solucionado en el proceso de realización de la prueba

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entidad/test | Descripción | Resultado esperado | Resultado real | Bugs encontrados |
| Contract/list-show.safe | Verificar que un cliente puede listar sus contratos y ver  sus detalles | Se espera que el sistema muestre los contratos y sus  detalles | El sistema muestra los contratos y sus detalles al cliente  correspondiente | Ninguno |
| Contract/create.safe | Verificar que un cliente puede crear un contrato introduciendo los valores que sea sin que aparezca un error 500 | Se espera que el sistema no lance ningún error 500 y salten todas las excepciones correctamente, además de crear un contrato al  introducir datos correctos | El sistema no lanza errores 500 y muestra todas las excepciones correctamente, además de crear correctamente el contrato al  introducir los datos correctos | Ninguno |
| Contract/update.safe | Verificar que un cliente puede actualizar un contrato sin que aparezca ningún error 500 | Se espera que en el caso positivo se actualice correctamente un contrato y en el caso negativo aparezca una  excepción | Se ha encontrado un error 500 el cual se ha arreglado y el software a actuado como debería | Se ha encontrado un error 500 ya que al introducir un código que ya existía este devolvía una pantalla de panic |
| Contract/publish.safe | Verificar que un cliente puede publicar un contrato sin que aparezca ningún error 500 | Se espera que en el caso positivo se publicar correctamente un contrato y en el caso negativo aparezca una  excepción | Se ha encontrado un error 500 el cual se ha arreglado y el software a actuado como debería | Se ha encontrado un error 500 ya que al introducir un código que ya existía este devolvía una pantalla de panic |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Contract/delete.safe | Verificar que un cliente puede eliminar un contrato  sin recibir errores | Se espera que un cliente elimine un contrato suyo | El cliente consigue eliminar correctamente un  contrato | Ninguno |
| Contract/create.hack | Verificar que un cliente no puede crear un contrato asociado a un proyecto no publicado | Se espera que la pagina ignore el intento de hackeo del cliente y complete la operación con el  proyecto correcto | La pagina ignora el intento y consigue completar la operación sin problemas | Ninguno |
| Contract/show-delete-update- publish.hack | Verificar que un cliente no puede cambiar datos que no se pueden cambiar, como el proyecto asignado a un contrato, que no pueda modificar y eliminar contratos de otro cliente | Se espera que la pagina lance un error lance un error 500 al intentar acceder a contratos que no son del cliente y que ignore al cliente en el intento de asignar un proyecto no  publicado | La pagina responde correctamente a todos los intentos de hackeo del cliente | Ninguno |
| ProgressLogs/create.safe | Verificar que un cliente puede crear un acta de progreso introduciendo los valores que sea sin que aparezca un error 500 | Se espera que el sistema no lance ningún error 500 y salten todas las excepciones correctamente, además de crear un acta de progreso al  introducir datos correctos | El sistema no lanza errores 500 y muestra todas las excepciones correctamente, además de crear correctamente el acta de progreso al  introducir los datos correctos | Ninguno |
| ProgressLogs/delete.safe | Verificar que un cliente puede eliminar un acta de progreso sin recibir  errores | Se espera que un cliente elimine un acta de progreso suyo | El cliente consigue eliminar correctamente un acta de progreso | Ninguno |
| Contract/list-show.safe | Verificar que un cliente puede listar su acta de progreso y ver sus detalles | Se espera que el sistema muestre el acta de progreso y sus detalles | El sistema muestra el acta de progreso y sus detalles al cliente  correspondiente | Ninguno |
| Contract/update.safe | Verificar que un cliente puede actualizar un acta de progreso sin que aparezca ningún error 500 | Se espera que en el caso positivo se actualice correctamente un acta de progreso y en el caso negativo aparezca una excepción | Se ha encontrado un error 500 el cual se ha arreglado y el software ha actuado como debería | Se ha encontrado un error 500 ya que al introducir un código que ya existía este devolvía una pantalla de panic |

**8**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Contract/publish.safe | Verificar que un cliente puede publicar un acta de progreso sin que aparezca ningún error 500 | Se espera que en el caso positivo se publicar correctamente un acta de progreso y en el caso negativo aparezca una  excepción | Se ha encontrado un error 500 el cual se ha arreglado y el software ha actuado como debería | Se ha encontrado un error 500 ya que al introducir un código que ya existía este devolvía una pantalla de panic |
| ProgressLogs/progressLogs.hack | Verificar que un cliente no puede crear, modificar, publicar, ver o eliminar actas de progreso de otro cliente o introducir valores que no debe | Se espera que en el caso de intentar acceder a actas de otro cliente salga un error 500 y que si intenta modificar datos incorrectos el sistema ignore al  cliente o aparezcan excepciones | El sistema actúa como debería | Ninguno |

## Test de rendimiento

En esta parte se evalúa el rendimiento del sistema de contratos y actas de progreso a partir de los análisis hechos usando el csv generado por el tester#replayer al ejecutarlo. Se proporcionarán gráficas y se calcularán intervalos de confianza del 95% para los tiempos dados realizando un contraste de hipótesis, también con un intervalo de confianza del 95% para determinar la diferencia de usar o no índices.

### Grafica de entidad sin índices:

**Grafica con entidades con índices:**

**Comparación de resultados:**

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

**Prueba z:**

|  |  |  |
| --- | --- | --- |
| Prueba z para medias de dos muestras | |  |
|  |  |  |
|  | *Before* | *After* |
| Media | 20,3225232 | 20,6723742 |
| Varianza (conocida) | 4227,01392 | 3317,16556 |
| Observaciones | 271 | 271 |
| Diferencia hipotética de las medias | 0 |  |
| z | -0,06630735 |  |
| P(Z<=z) una cola | 0,47356657 |  |
| Valor crítico de z (una cola) | 1,64485363 |  |
| Valor crítico de z (dos colas) | 0,94713313 |  |
| Valor crítico de z (dos colas) | 1,95996398 |  |

Hemos obtenido un valor critico de z de 0,947 para un nivel de significancia de 0.95, esto lo usaremos para saber si los valores entre las pruebas son significativos.

De acuerdo a la metodología del curso, si el p-valor esta en el intervalo (alfa, 1.00] esto indica que los cambios no resultaron en mejoras relevantes, dado que nuestro valor z si está en ese intervalo eso quiere decir que no han sido significantes en el rendimiento, esto es debido a que la mayoría de las llamadas a base de datos son simples y no usan clausulas where complejas, además de ser la mayoría sobre el atributo id, el cual ya tiene un índice implícito, pudiendo asegurar un nivel de confianza de pruebas del 95%

### Cobertura:

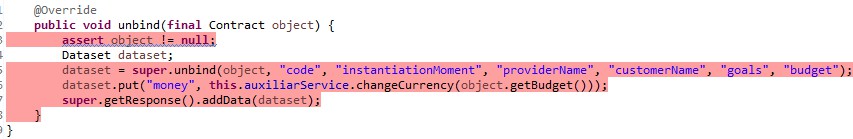
En esta sección hablaremos de la cobertura que le han dado las pruebas al código generado indicando los porcentajes de cobertura dados en cada archivo

### Cobertura de contracts:

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Se tiene una cobertura de casi el 60%, debido a la baja cobertura del deleteService y el publishService, esto se debe a la función unbind la cual no se ejecuta ya que no se produce ningún caso en el que el contrato que se va a eliminar sea nulo, ocurre lo mismo en el ClientContractPublishService



Si quitáramos es función alcanzaríamos una cobertura probablemente superior al 65%

### Cobertura de Progress Logs:

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

Se tiene una cobertura del 33.3% debido a que el archivo ListService no se utiliza, en cambio se usa el ListAllService el cual tiene una cobertura muy alta, este primer archivo continua ahí debido a decisiones mal tomadas al principio del sprint 3.

Una vez más tenemos el mismo problema con el deleteService y el publishService, debido a que la función unbind no se utiliza genera una mala cobertura

# Conclusiones

Este entregable individual ha sido el entregable más duro debido a la complejidad y poca familiarización con el entorno, sin embargo, se han podido realizar a tiempo todas las tareas, tanto obligatorias como opcionales.

# Bibliografía

Intencionalmente en blanco.