Universidad de Sevilla

Escuela Técnica Superior de Ingeniería Informática

**Documentación de la entrega D04**

**Testing**



Grado en Ingeniería Informática – Ingeniería del Software  
 Diseño y Pruebas 2

Curso 2023 – 2024

|  |  |
| --- | --- |
| **Fecha** | **Versión** |
| 03/03/2024 | v1r1 |

|  |  |  |
| --- | --- | --- |
| **Grupo de prácticas: G6-64** | | |
| **Autores por orden alfabético** | **Rol** | **Correo electrónico** |
| Aguayo Orozco, Sergio - 25604244T | Desarrollador | ahydul1@gmail.com |
| García Lama, Gonzalo - 47267072W | Desarrollador, Tester | gongarlam@alum.us.es |
| Huecas Calderón, Tomás - 17476993Y | Desarrollador | tomhuecal@alum.us.es |
| Fernández Pérez, Pablo - 54370557Y | Desarrollador,  Analista | pablofp.33@gmail.com |
| Youssafi Benichikh, Karim - 28823709V | Desarrollador, operador, manager | karyouben@alum.us.es |

## Repositorio: <https://github.com/karyouben/Acme-SF-D04>

**Índice de contenido**

[1. Resumen ejecutivo 3](#_Toc158967746)

[2. Control de versiones 4](#_Toc158967747)

[3. Introducción 5](#_Toc158967748)

[4. Contenido 6](#_Toc158967749)

[5. Conclusiones 6](#_Toc158967750)

[6. Bibliografía 18](#_Toc158967751)

# 1. Resumen ejecutivo

El documento de pruebas está dividido en dos apartados. El primero está centrado en pruebas funcionales y el segundo está centrado en pruebas de rendimiento.

# 2. Control de versiones

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** |
| 26/05/2024 | v1r0 | Creado el documento |
| 27/05/2024 | v2r0 | Terminado el documento |
|  |  |  |
|  |  |  |

# 3. Introducción

A continuación, mostramos los resultados de las pruebas. El primer apartado incluye un listado con los casos de prueba implementados, agrupados por implementación, con descripción sobre su efectividad a la hora de encontrar errores.

El segundo apartado incluye información sobre el rendimiento de la aplicación. Gráficas e intervalos de confianza 95% tomados en dos ordenadores distintos, además de un contraste de hipótesis de confianza respecto a qué ordenador es más potente.

# 4. Contenido

Se toma la decisión de no implementar ningún índice en el framework ya que este crea los índices por ids y atributos de navegación automáticamente

Algunas cosas pueden salir en amarillo como el assert object != null o validaciones authenticated que ya lo haga el framework por detrás.

Al mergear todo al main puede ser que los csv den conflictos o se hayan cambiado etc… pero en mi rama tomhuecal va todo perfecto. Se toma la decisión de eliminar los tests de los demás compañeros en la rama personal para agilizar el proceso de testing.

https://github.com/karyouben/Acme-SF-D04/tree/tomhuecal

**Pruebas Funcionales**

**Contract**

|  |  |
| --- | --- |
| Create.safe | Se inicia sesión como client1 y se prueba a crear contratos de todas las formas que permite el sistema, comprobando los límites y los mensajes de error. Comprobando restricciones de reglas de negocio como que, para el proyecto seleccionado en el contrato, el coste de este sea inferior al total cost del proyecto. |
| Delete.safe | Se inicia sesión como client1 y se prueba a eliminar un contrato, comprobamos restricción de que no se pueda eliminar un contrato que ya tiene progresos publicados. |
| List.safe | Se inicia sesión como client1 y se listan sus contratos. No se detecta ningún error. |
| Update.safe | Se inicia sesión como client1 y se prueba a actualizar los datos de un contrato de todas las formas que permite el sistema, comprobando los límites y los mensajes de error. **Se detecta** un error, cuando el proyecto seleccionado lo cambias a uno vacio salta exception ya que para que dejar actualizar el contrato se necesita consultar el total cost del proyecto, pero al ser nulo salta NullPointerException. Solución propuesta: Después de hablarlo con el cliente se toma la decisión de que el proyecto no sea modificable en el update. |
| Publish.safe | Se inicia sesión como client1 y se prueba a publicar un contrato. Se prueba a publicar el contrato cuando no tengo algún progreso en modo borrador. No se detecta ningún error. |
| Show.safe | Se inicia sesión como client1, se listan sus contratos y se muestra un contrato del listado. No se detecta ningún error. |
| Delete.hack | Nos metemos en un contrato del client1 y se cambia la id del formulario para que corresponda a la id de un contrato de otro cliente y se prueba a eliminar y no deja. |
| List.hack | Se prueba a listar contratos sin tener rol. No deja |
| Update.hack | Nos metemos en un contrato del client1 y se cambia la id del formulario para que corresponda a la id de un contrato de otro cliente y se prueba a hacer update y no deja. |
| Publish.hack | Se prueba a publicar un contrato sin rol, con rol distinto, y con el el rol pertinente del cliente, pero no deja publicarlo porque no tiene progress log sin asociados sin publicar. |
| Show.hack | Se hace show con id 282 cuando no está con algún rol o no es la misma cuenta del rol y no deja mostrar el contrato. |

**Progress log**

|  |  |
| --- | --- |
| Create.safe | Se inicia sesión como client1 y se prueba a crear un progress log asociado a un contrato de todas las formas posibles. No se detecta error |
| Delete.safe | Se intenta eliminar un progress log y no hay problema. Se recuerda que los progress log así como los contratos solo pueden ser eliminador o updateados cuando no han sido publicados. |
| List.safe | Se inicia sesión como client1 todos los progress log. No se detecta fallo |
| list-by-contract.safe | Se inicia sesión como client1 y hace show de un contrato que tenga progress log asociados. Al darle a la opción de list progress log se muestras todos los progress log asociados a ese contrato. No se detecta fallo |
| Publish.safe | Se inicia sesión como un client1 y se publica un progress log. |
| Show.safe | Se inicia sesión como client1 y se listas los progress log. Al clickear uno hace un show con la id de ese progress log no se detectan fallos. |
| Update.safe | Se inicia sesión como client1 y se prueba a actualizar un progress log de todas las formas posibles (campos vacíos, límite de caracteres etc) |
| List.hack | Se hace /progress/list cuando no esta logeado, o con otro rol distinto y no muestra la lista de los progresos. |
| List-by-contract.hack | Se hace client/progress/list-by-contract?contractId=438 Para usuario anonimo y otros roles distintos. No se detecta problema |
| Publish.hack | Se hace client/progress/publish?id=438 sin logear y para gente con distintos roles y no deja publicar el progress log. |
| Delete.hack | Se prueba a un progress log, para ello en el formulario se cambia la id por otro progress distinto. No deja |
| Show.hack | Se prueba a consultar un progress log sin logear, no deja. |
| Update.hack | Se prueba a actualizar un progress log probando limites en el formulario y cambiando la id del formulario por otro progress log que no pertenece al usuario logeado. No deja |

En resumen se ha abordado una gran variedad de casos posibles, en el coverage de las pruebas podemos observar mas de un 90 % de coverage en todos los servicios relacionados a los requisitos que ha pedido el cliente para el student 2.

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamente con confianza media

**Pruebas de Rendimiento**

**PC A**

**Gráfico, Gráfico en cascada

Descripción generada automáticamente**

**PC B**

Gráfico

Descripción generada automáticamente

El PC B por lo general tiene tiempos más bajos, observamos que la request que tarda más tiempo es listar todos los contratos esto es lógico ya que es cuando mas datos se tienen que fetchear y por lo tanto mas tiempo.

# **Interfaz de usuario gráfica, Aplicación, Tabla, Excel Descripción generada automáticamente**

Interfaz de usuario gráfica, Aplicación, Tabla, Excel

Descripción generada automáticamente

# El PCA tiene intervalo de confianza 95% (23.75, 35.5) y el PCB tiene intervalo (20, 31). Son intervalos aceptables.

A continuación, calculamos la hipótesis de contraste con 95% confianza para intentar averiguar que ordenador es más potente:

# 

# **Interfaz de usuario gráfica, Texto, Aplicación, Tabla Descripción generada automáticamente**

El valor de P sale razonable. En el caso PCB, es mejor debido a que se ha probado con un ordenador con un procesador más potente y tiene una media de tiempos menor.

# 5. Conclusiones

Este documento guarda las pruebas que se han realizado durante este entregable, las cuales han servido para encontrar algún errores/bugs en el código y calcular el rendimiento de este comparándolo con dos ordenadores.

# 6. Bibliografía

Intencionadamente en blanco