

Universidad de Sevilla

Escuela Técnica Superior de Ingeniería Informática

Testing Report



Grado en Ingeniería Informática – Ingeniería del Software

Diseño y Pruebas 2.

Curso 2023 – 2024

Fecha	Versión
26/05/2024	v1.0.0

Grupo: C1.011		
Autores por orden alfabético	Rol	Descripción del rol
Abouri, Mohamed - Y7156458E	Developer	Persona encargada de desarrollar el código.
Cabello Ranea, Rafael - 32094496C	Operador	Encargado de las tareas de campo, de las instalaciones y del mantenimiento de los sistemas de la empresa.
Calderón Rodríguez, Francisco Javier - 78233219F	Project Manager	Persona encargada de tomar decisiones de diseño y vigilar el correcto desarrollo
Delgado Pallares, David - 29519510E	Tester	Persona encargada de realizar pruebas sobre el código.
Ramírez Lara, Iván- 77852382G	Developer	Persona encargada de desarrollar el código.

	Diseño y Pruebas 2 Testing Report
	Control de versiones

Control de Versiones

Fecha	Versión	Descripción
26/05/2024	v1.0.0	Desarrollo de la primera versión.

	<p>Diseño y Pruebas 2 Testing Report</p>
---	--

Índice de contenido

1. Introducción
2. Testeo funcional
3. Testeo de rendimiento
4. Cobertura

	<div>Diseño y Pruebas 2 Testing Report</div>
---	--

1. Introducción

Este documento presenta dos capítulos esenciales: pruebas funcionales, con un listado de casos de prueba organizados por característica y su efectividad en detectar errores, y pruebas de rendimiento, con gráficos e intervalos de confianza del 95% para el tiempo de respuesta en dos computadoras, además de un análisis para determinar cuál es más potente. Estos capítulos son cruciales para evaluar la calidad y el rendimiento del software.

2. Testeo funcional

En esta sección se detallarán todas las pruebas que hemos realizado para testear el funcionamiento de nuestro software.

Descripción de la prueba: Se detallará cada prueba realizada, incluyendo los pasos específicos llevados a cabo.

Resultado esperado: Se especificará cuál era el comportamiento o el resultado que se esperaba obtener con la prueba.

Resultado real: Se presentará el resultado obtenido tras la ejecución del test, ya corregido después de encontrar y solucionar cualquier bug que afectará el código.

	Diseño y Pruebas 2 Testing Report
---	--------------------------------------

List

Entidad	Descripción	Resultado Esperado	Resultado Real	Bugs detectados
contract safe	Verificar que un cliente puede listar sus contratos.	El sistema deberá mostrar los contratos asociados al cliente	El sistema ha mostrado los contratos asociados al cliente	Ninguno
contract hack	Verificar que un rol no cliente no puede listar contratos.	El sistema no debe mostrar los contratos, usando un rol no cliente, dando un error de pánico	El sistema ha dado un error 500 de no autorizado en cada rol no cliente.	Ninguno
progress log safe	Comprobar que un cliente puede listar los registros de progreso de los contratos.	El sistema deberá mostrar los registros de progreso asociados a un contrato del cliente	El sistema ha mostrado todos los registros de progreso asociados a cada uno de los contratos probados	Ninguno
progress log hack	Comprobar que un rol no cliente no puede listar registros de progreso asociados a un contrato determinado. Ni otro cliente puede listar registros de progreso de un contrato de otro cliente.	El sistema no debe mostrar los registros de progreso asociados a un contrato, usando un rol no cliente, dando un error de pánico	El sistema ha dado un error 500 de no autorizado en cada rol no cliente. También ha dado error cuando se ha intentado acceder a los registros de progreso de un contrato con un cliente diferente.	Ninguno

	Diseño y Pruebas 2 Testing Report
---	--------------------------------------

Show

Entidad	Descripción	Resultado Esperado	Resultado Real	Bugs detectados
contract safe	Verificar que un cliente puede ver toda la información de un contrato.	El sistema deberá mostrar la información de los contratos asociados al cliente	El sistema ha mostrado la información de los contratos asociados al cliente	Ninguno
contract hack	Verificar que un rol no cliente no puede ver la información de los contratos.	El sistema no debe mostrar la información de contratos, usando un rol no cliente, dando un error de pánico	El sistema ha dado un error 500 de no autorizado en cada rol no cliente.	Ninguno
progress log safe	Comprobar que un cliente puede ver la información asociada a los registros de progreso.	El sistema deberá mostrar la información de los registros de progreso asociados al contrato del cliente.	El sistema ha mostrado toda la información de los registros de progreso elegidos y que están asociados a contratos del cliente.	Ninguno
progress log hack	Comprobar que un rol no cliente no puede ver la información asociada a los registros de progreso.	El sistema no debe mostrar los registros de progreso, usando a un rol no cliente, dando un error de pánico	El sistema ha dado un error 500 de no autorizado en cada rol no cliente. Al intentar acceder a la información de un registro de progreso, también lo ha dado si el registro de progreso no pertenece a un contrato del cliente utilizado.	Ninguno

	Diseño y Pruebas 2 Testing Report
---	--------------------------------------

Create

Entidad	Descripción	Resultado Esperado	Resultado Real	Bugs detectados
contract safe	Verificar que un cliente puede crear contratos de forma correcta.	El sistema deberá permitir crear contratos válidos.	El sistema ha dejado crear contratos con datos válidos.	Ninguno
contract hack	Verificar que un rol no cliente no puede acceder a la creación de contratos.	El sistema debe de lanzar un error de pánico si un rol no cliente intenta acceder a la URL o si se hace una petición sin ser cliente.	El sistema ha dado un error 500 de no autorizado en cada rol no cliente.	Ninguno
progress log safe	Comprobar que un cliente puede crear registros de progreso de forma correcta.	El sistema deberá permitir crear registros de progreso válidos.	El sistema ha permitido crear registros de progreso válidos.	Ninguno
progress log hack	Revisar que un rol no cliente no puede acceder a la creación de registros de progreso.	El sistema debe de lanzar un error de pánico si un rol no cliente intenta acceder a la URL o si se hace una petición sin ser cliente.	El sistema ha dado un error 500 de no autorizado en cada rol no cliente. Al intentar acceder a la información de un registro de progreso. También lo ha dado si el registro de progreso no pertenece a un contrato del cliente utilizado	Ninguno

Publish

Entidad	Descripción	Resultado	Resultado Real	Bugs detectados
---------	-------------	-----------	----------------	-----------------

	Diseño y Pruebas 2 Testing Report
---	--------------------------------------

		Esperado		
contract safe	Verificar que un cliente puede publicar contratos válidos de forma correcta.	El sistema deberá permitir publicar contratos válidos.	El sistema ha dejado publicar contratos con datos válidos.	Ninguno
contract hack	Verificar que un rol no cliente u otro cliente no puede acceder a la publicación de contratos.	El sistema debe de lanzar error de pánico si un rol no cliente intenta acceder a la URL, si se hace una petición sin ser cliente o si se hace desde un cliente diferente	El sistema ha dado un error 500 de no autorizado en cada rol no cliente. También ha dado error 500 si un cliente intentaba acceder a un contrato de otro cliente	Ninguno
progress log safe	Comprobar que un cliente puede publicar un registro de progreso de forma correcta.	El sistema deberá permitir publicar registros de progreso válidos.	El sistema ha permitido publicar los registros de progreso elegidos.	Ninguno
progress log hack	Comprobar que un rol no cliente no puede acceder a la publicación de registros de progreso.	El sistema debe de lanzar un error de pánico si un rol no cliente intenta acceder a la URL o si se hace una petición sin ser cliente.	El sistema ha dado un error 500 de no autorizado en cada rol no cliente. Al intentar acceder a la información de un registro de progreso. También lo ha dado si el registro de progreso no pertenece al cliente utilizado	Ninguno

Delete

Entidad	Descripción	Resultado Esperado	Resultado Real	Bugs detectados
---------	-------------	--------------------	----------------	-----------------

	<p align="center">Diseño y Pruebas 2 Testing Report</p>
---	---

contract safe	Verificar que un cliente puede borrar contratos de forma correcta.	El sistema deberá permitir eliminar contratos que no estén publicados.	El sistema ha dejado eliminar contratos no publicados.	Ninguno.
contract hack	Verificar que un rol no cliente no puede acceder a la eliminación de contratos.	El sistema debe de lanzar un error de pánico si un rol no cliente intenta acceder a la URL o si se hace una petición sin ser cliente.	El sistema ha dado un error 500 de no autorizado en cada rol no cliente y en cliente no autorizados.	Ninguno
progress log safe	Comprobar que un cliente puede eliminar registros de progreso de forma correcta.	El sistema deberá permitir eliminar registros de progreso no publicados.	El sistema ha permitido eliminar registros de progreso válidos.	Ninguno
progress log hack	Revisar que un rol no cliente no puede acceder a la eliminación de registros de progreso.	El sistema debe de lanzar un error de pánico si un rol no cliente intenta acceder a la URL o si se hace una petición sin ser cliente.	El sistema ha dado un error 500 de no autorizado en cada rol no cliente. Al intentar acceder a la información de un registro de progreso. También lo ha dado si el registro de progreso no pertenece al cliente utilizado	Ninguno

Update

Entidad	Descripción	Resultado Esperado	Resultado Real	Bugs detectados
contract safe	Verificar que un cliente puede actualizar contratos	El sistema deberá permitir actualizar contratos válidos.	El sistema ha dejado actualizar contratos con datos válidos.	Ninguno

	<p align="center">Diseño y Pruebas 2 Testing Report</p>
---	---

	válidos de forma correcta.			
contract hack	Verificar que un rol no cliente u otro cliente no puede acceder a la actualización de contratos.	El sistema debe de lanzar error de pánico si un rol no cliente intenta acceder a la URL, si se hace una petición sin ser cliente o si se hace desde un cliente diferente	El sistema ha dado un error 500 de no autorizado en cada rol no cliente. También ha dado error 500 si un cliente intentaba acceder a un contrato de otro cliente	Ninguno
progress log safe	Comprobar que un cliente puede actualizar un registro de progreso de forma correcta.	El sistema deberá actualizar registros de progresos válidos.	El sistema ha permitido actualizar los registros de progreso elegidos.	Ninguno
progress log hack	Comprobar que un rol no cliente no puede acceder a la actualización de registros de progreso.	El sistema debe de lanzar un error de pánico si un rol no cliente intenta acceder a la URL o si se hace una petición sin ser cliente.	El sistema ha dado un error 500 de no autorizado en cada rol no cliente. Al intentar acceder a la información de un registro de progreso. También lo ha dado si el registro de progreso no pertenece al cliente utilizado	Ninguno

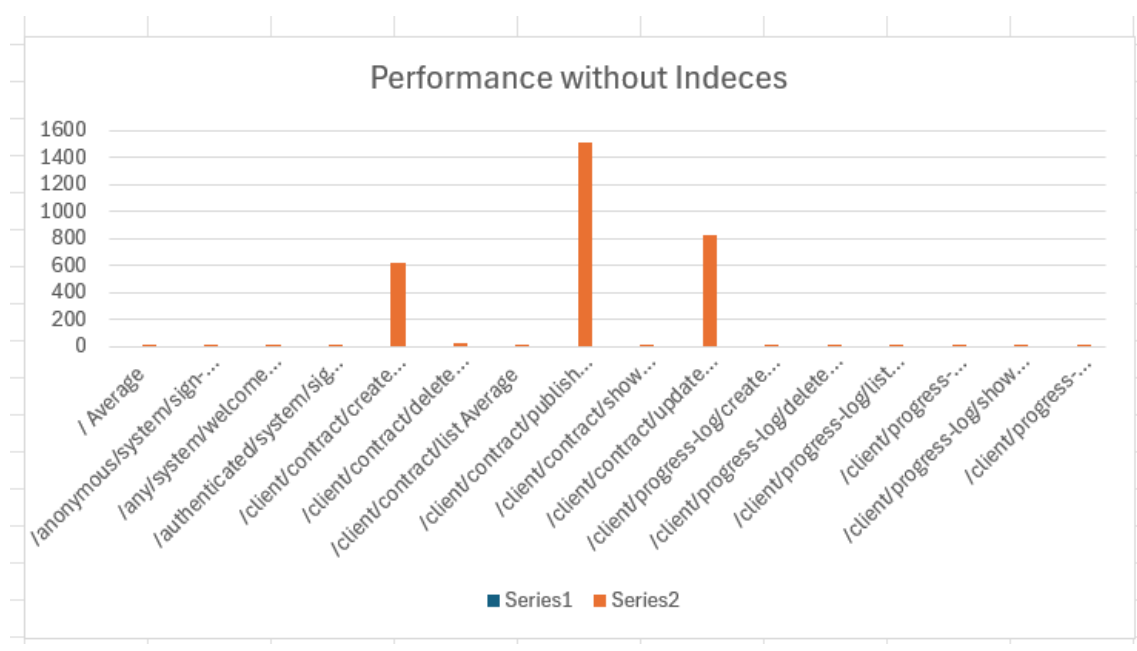
3. Testeo de rendimiento

En este capítulo, se evaluará el rendimiento de nuestro proyecto mediante pruebas detalladas que analizan el tiempo de respuesta al atender las solicitudes en nuestras pruebas funcionales. Utilizaremos una muestra compuesta por los resultados obtenidos por hacer replay desde coverage, sobre las entidades con índices y sin estos.

Para realizar el análisis de la prueba, presentaremos gráficos que ilustran los resultados obtenidos. Además, calcularemos intervalos de confianza del 95% para los tiempos de respuesta medidos y realizaremos un contraste de hipótesis, también con un intervalo de confianza del 95%, para determinar la diferencia de hacer replay sobre entidades con índices o entidades sin estos.

Muestra sobre resultados desde Coverage.

Gráficas de entidades sin índices:




Gráficas de entidades con índices:



Comparación de resultados:

Before				After			
Mean	77.5397317			Mean	79.36911198		
Standard Error	11.25097723			Standard Error	11.34833963		
Median	4.82445			Median	4.8145		
Mode	2.2764			Mode	2.3991		
Standard Deviation	340.8879004			Standard Deviation	343.8378364		
Sample Variance	116204.5607			Sample Variance	118224.4577		
Kurtosis	27.73715662			Kurtosis	28.24477372		
Skewness	5.168724819			Skewness	5.18407187		
Range	2622.7984			Range	2920.1792		
Minimum	1.281			Minimum	1.308		
Maximum	2624.0794			Maximum	2921.4872		
Sum	71181.4737			Sum	72860.8448		
Count	918			Count	918		
Confidence Level(95.0%)	22.08065416			Confidence Level(95.0%)	22.27173314		
Interval (ms)	55.45907754	99.62038586		Interval (ms)	57.09737884	101.6408451	


	Diseño y Pruebas 2 Testing Report
---	--------------------------------------

Prueba Z:

A	B	C
z-Test: Two Sample for Means		
	<i>Before</i>	<i>After</i>
Mean	77.5397317	79.369112
Known Variance	116204.5607	118224.458
Observations	918	918
Hypothesized Mean Difference	0	
z	-0.114477372	
P(Z<=z) one-tail	0.454429692	
z Critical one-tail	1.644853627	
P(Z<=z) two-tail	0.908859383	
z Critical two-tail	1.959963985	

En nuestras pruebas de rendimiento, hemos obtenido un valor crítico de z de **0.908859** para un nivel de significancia (α) de **0.95**. Este valor se utiliza para determinar si las diferencias en los tiempos de respuesta entre las dos pruebas son estadísticamente significativas.

De acuerdo con nuestra metodología, si el p-valor se encuentra en el intervalo (α , 1.00], esto indica que los cambios no resultaron en mejoras relevantes; es decir, aunque los tiempos de muestra sean diferentes, globalmente son equivalentes. Dado que nuestro valor crítico de z está en este intervalo, concluimos que no hay una mejora significativa en el rendimiento al comparar los tiempos de respuesta de las pruebas. En resumen, las diferencias observadas no son lo suficientemente grandes como para considerarse estadísticamente relevantes con el nivel de confianza del 95%.

	<div>Diseño y Pruebas 2 Testing Report</div>
---	--

Conclusión

Una vez mostrado los resultados, podemos ver que los servicios que más tiempo consumen con mucha diferencia son: Contract Create, Contract Publish y Contract Update. Esto se debe a que estos servicios consumen una api externa para el cambio de divisa, que es muy lenta. Además los resultados de las consultas a la api no se guardan en una caché. Esto hace que el rendimiento de estos servicios sea muy lento.

Cobertura

En esta sección, analizaremos la cobertura de código lograda con nuestros tests. La cobertura de código indica qué porcentaje del código fuente ha sido ejecutado durante las pruebas, ayudando a identificar áreas verificadas y posibles errores no detectados.

Discutiremos los métodos de medición, presentaremos los resultados obtenidos y evaluaremos la efectividad de nuestros tests. También identificaremos lagunas en la cobertura y propondremos estrategias para mejorarla, asegurando así mayor confiabilidad y robustez del software.

Cobertura registro de progreso:

acme.features.client.progress_log	87.9 %	910	125	1,035
> ClientProgressLogDeleteServi	63.3 %	107	62	169
> ClientProgressLogCreateServi	91.6 %	186	17	203
> ClientProgressLogPublishServ	91.9 %	181	16	197
> ClientProgressLogUpdateServ	91.7 %	177	16	193
> ClientProgressLogListService.j	92.4 %	121	10	131
> ClientProgressLogShowServic	96.3 %	103	4	107
> ClientProgressLogController.j	100.0 %	35	0	35

Como se puede apreciar, la cobertura sobre las features de la entidad registro de progreso es bastante buena, ya que todas las features implementadas tienen una cobertura de más del 90% excepto el delete.

Esto ocurre porque el delete cuenta con el método unbind, pero este no es nunca usado porque no es necesario para esta funcionalidad, si comentamos este código o lo borramos la cobertura ya superaría el 90% de porcentaje, es por esto que no estamos preocupados por que este no llegue al 90% de cobertura.

```


@Override
public void unbind(final ProgressLog object) {
    assert object != null;

    Dataset dataset;

    dataset = super.unbind(object, "recordId", "completeness", "comment", "registrationMoment", "responsible", "draftMode");
    dataset.put("masterId", object.getContract().getId());

    super.getResponse().addData(dataset);
}

```

	Diseño y Pruebas 2 Testing Report
---	--------------------------------------

Cobertura de contrato:

Con respecto a esta entidad, ocurre una cosa parecida a la comentada anteriormente, ya que la cobertura de todas las features es de mayor que el 90% exceptuando a delete que por el unbind no supera este porcentaje.

acme.features.client.contract	89.7 %	1,399	160	1,559
> ClientContractDeleteService.j	60.7 %	136	88	224
> ClientContractUpdateService.j	94.5 %	325	19	344
> ClientContractPublishService.j	95.7 %	397	18	415
> ClientContractCreateService.j	94.9 %	299	16	315
> ClientContractShowService.ja	91.5 %	161	15	176
> ClientContractListService.java	92.0 %	46	4	50
> ClientContractController.java	100.0 %	35	0	35

```

@Override
public void unbind(final Contract object) {
    assert object != null;

    int clientId;
    Collection<Project> projects;
    SelectChoices choices;
    Dataset dataset;

    clientId = super.getRequest().getPrincipal().getActiveRoleId();
    projects = this.repository.findManyProjectsByClientId(clientId);
    choices = SelectChoices.from(projects, "code", object.getProject());

    dataset = super.unbind(object, "code", "instantiationMoment", "provider", "customer", "goals", "budget", "draftMode");
    dataset.put("project", choices.getSelected().getKey());
    dataset.put("projects", choices);

    super.getResponse().addData(dataset);
}

```