



UNIVERSIDAD  
DE SEVILLA

# E2 Testing report D04

**Group:** C1.031

**Student #2**

**Name:** Nuñez Sanchez Juan  
**Email:** juanunsan2@alum.us.es

**Repository:** <https://github.com/alvarous/Acme-One-C1.031>

# Contenido

## Contenido

- Contenido ..... 3
- Resumen ejecutivo ..... 4
- Registro de cambios ..... 5
- Introducción ..... 6
- Testing Funcional ..... 7
- Test de Rendimiento ..... 9
- Conclusión ..... 11
- Bibliografía ..... 12

## Resumen ejecutivo

En este documento veremos todos los casos de prueba y nos daremos cuenta de que hemos encontrado algunos bugs en view de Contract, un bug en la implementación de la API y hemos detectado problemas en la base de datos debido a la insuficiencia de pruebas en la población de los datos que dificultaban la creación de estas pruebas.

Registro de cambios

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
1.0	Testing Report	Juan Nuñez Sanchez	26/05/2024

## Introducción

En este documento, se incluirán tanto los casos de prueba que hemos llevado a cabo como un análisis detallado de estos y una evaluación del rendimiento de nuestro proyecto en un ordenador de gama media.

## Testing Funcional

Primero comentaremos las diferentes pruebas realizadas para realizar testing funcional para las entidades contract y progressLogs creadas por el estudiante:

### CONTRACT.

**Listar:** Para verificar que la funcionalidad de listado funciona correctamente, iniciamos sesión como cliente y accedimos a la lista de contratos. Tras confirmar que los campos se muestran correctamente, dimos por concluida la prueba. Esta prueba no detectó ningún bug.

**Mostrar:** Para asegurarnos de que la función de visualización funciona adecuadamente, iniciamos sesión como cliente y accedimos a la lista de contratos para luego ingresar en uno de ellos. Tras verificar que los campos se muestran de manera correcta, dimos por finalizada la prueba. Esta prueba no detectó ningún bug.

**Crear:** Para comprobar que la creación de un contrato funciona correctamente, probamos diferentes posibilidades: intentamos crearlo vacío, crearlo con todos los campos válidos excepto uno vacío, y crear un contrato ya existente. También probamos cada campo de entrada uno por uno con diferentes valores no válidos. Detectamos un bug que impedía la creación de un contrato vacío y provocaba un error 500, el cual ha sido solucionado.

**Eliminar:** Para verificar que la eliminación funciona correctamente, accedimos a un contrato sin publicar y lo eliminamos. No se pudo probar la función de "unbind" ya que no es posible eliminar un contrato de un contrato ya publicado.

**Actualizar:** Para asegurar el correcto funcionamiento de la actualización, repetimos exactamente las mismas pruebas realizadas para la creación de un contrato. Estas pruebas no detectaron ningún bug.

**Publicar:** Para verificar la funcionalidad de publicación de contratos, repetimos las mismas pruebas que en la creación y actualización. Además, probamos cada campo de entrada uno por uno con diferentes valores no válidos.

### Progress Logs.

**Listar:** Para verificar que la funcionalidad de listado funciona correctamente, iniciamos sesión como cliente y accedimos a la lista de contratos, entramos en uno de ellos y accedemos a sus progress logs. Tras confirmar que los campos se muestran correctamente, dimos por concluida la prueba. Esta prueba no detectó ningún bug.

**Mostrar:** Para asegurarnos de que la función de visualización funciona adecuadamente, iniciamos sesión como cliente y accedimos a la lista de contratos, entramos en uno de ellos y accedemos a sus progress logs y accedemos a uno de ellos. Tras verificar que los campos se muestran de manera correcta, dimos por finalizada la prueba. Esta prueba no detectó ningún bug.

**Crear:** Para comprobar que la creación de un progress log funciona correctamente, probamos diferentes posibilidades: intentamos crearlo vacío, crearlo con todos los campos válidos excepto uno vacío, y crear un contrato ya existente. También probamos cada campo de entrada uno por uno con diferentes valores no válidos. No se detectó ningún bug.

**Eliminar:** Para verificar que la eliminación de un progress log funciona correctamente, accedimos a un progress log no publicado y lo eliminamos. No se pudo probar la función de "unbind" ya que no es posible eliminar un progress log ya publicado.

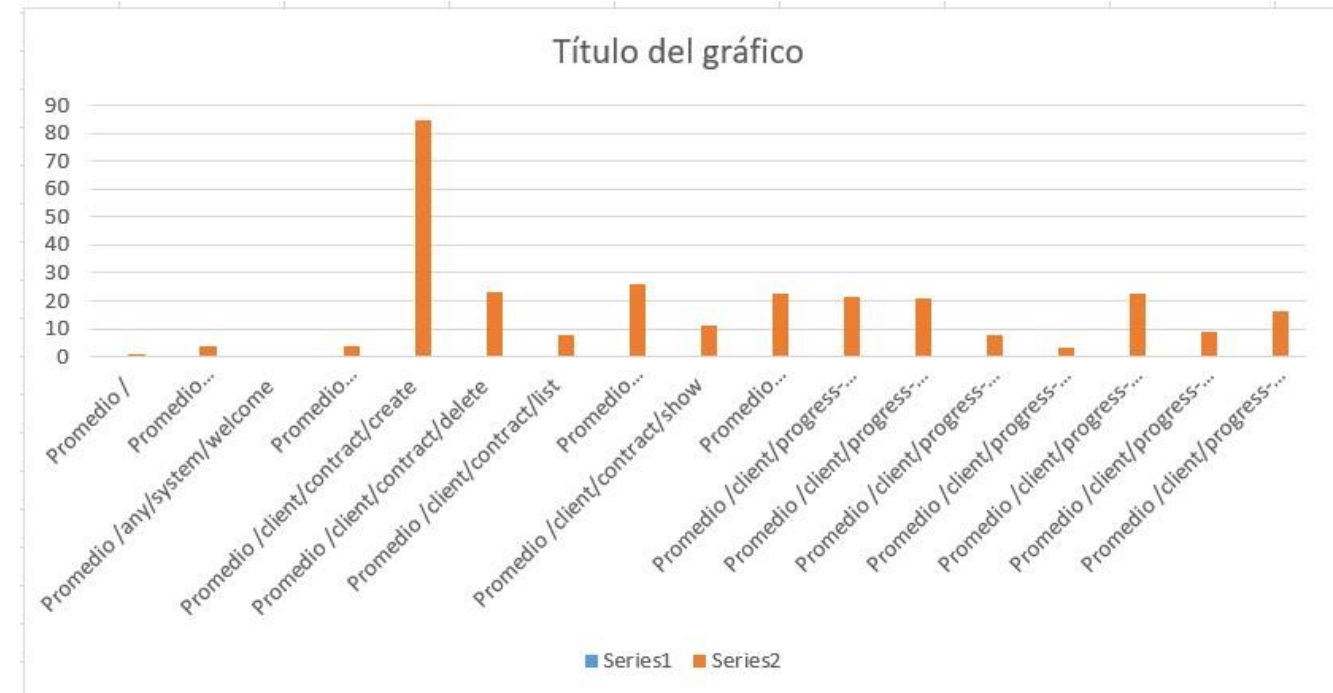
**Actualizar:** Para asegurar el correcto funcionamiento de la actualización, repetimos exactamente las mismas pruebas realizadas para la creación de un progress log. Estas pruebas no detectaron ningún bug.

**Publicar:** Para verificar la funcionalidad de publicación de progress logs, repetimos las mismas pruebas que en la creación y actualización.



# Test de Rendimiento.

Para el test de rendimiento se han registrado los tiempos promedio que ha tomado la aplicación en responder a cada solicitud. Este gráfico presenta esos resultados.



Podemos observar que el principal problema de rendimiento lo encontramos en el create de Contract. Esto se debe a que es en este servicio donde se han realizado más llamadas y llevado a cabo llamadas a la API para el cambio de moneda. Se ha intentado mejorar este rendimiento pero ha sido poco efectivo

Tras la implementación de índices en las llamadas a la base de datos se realizó un análisis de las estadísticas antes y después, resultando los valores de la siguiente manera:

Columna1				Columna1			
Media	15,0664364			Media	12,57645075		
Error típico	2,55003633			Error típico	1,780023495		
Mediana	5,2388			Mediana	5,6274		
Moda	0,385			Moda	0,3121		
Desviación estándar	55,9266629			Desviación estándar	38,46663309		
Varianza de la muestra	3127,79162			Varianza de la muestra	1479,681862		
Curtosis	145,588408			Curtosis	274,9305086		
Coefficiente de asimetría	10,7351176			Coefficiente de asimetría	15,23313004		
Rango	906,5964			Rango	736,5599		
Mínimo	0,1729			Mínimo	0,1788		
Máximo	906,7693			Máximo	736,7387		
Suma	7246,9559			Suma	5873,2025		
Cuenta	481			Cuenta	467		
Nivel de confianza(95,0%)	5,01061353			Nivel de confianza(95,0%)	3,497866708		
intervalo(ms)	10,0558228	20,07704992		intervalo(ms)	9,078584041	16,0743175	
intervalo(s)	0,01005582	0,02007705		intervalo(s)	0,009078584	0,01607432	

Finalmente comprobamos el intervalo de confianza, resultando en un correcto 0,42. Mostrando que el rendimiento no ha obtenido mucha mejora y que los valores son muy parecidos con y sin índices. Esto se puede justificar con que no haya suficientes llamadas en la que los índices se vean involucrados para que hagan efecto:

Prueba z para medias de dos muestras		
	<i>BEFORE</i>	<i>AFTER</i>
Media	15,0664364	12,5764507
Varianza (conocida)	3127,79162	1479,68
Observaciones	481	467
Diferencia hipotética de las medias	0	
z	0,80067717	
P(Z<=z) una cola	0,21165928	
Valor crítico de z (una cola)	1,64485363	
Valor crítico de z (dos colas)	0,42331856	
Valor crítico de z (dos colas)	1,95996398	

## Conclusión

En conclusión, aunque realizar pruebas puede ser una tarea tediosa y repetitiva, hacerlo de manera sistemática es fundamental para detectar errores y fallos en el proyecto. Además, el análisis de rendimiento nos permite ver claramente cómo la introducción de índices en la base de datos mejora el funcionamiento, aunque para haberlo notado más en este caso hubiera sido necesario que los índices tuvieran más peso en las llamadas. En general, estas actividades son muy beneficiosas y esenciales durante el desarrollo de un producto de software.

## Bibliografía

Intencionalmente en blanco.