



# Testing Report

C3.017

<https://github.com/vicgrabru/Acme-SF-D04>

Francisco de Asís Rosso Ramírez  
frarosram@alum.us.es

# Tabla de contenidos

<b>Tabla de contenidos.....</b>	<b>1</b>
<b>Resumen ejecutivo.....</b>	<b>2</b>
<b>Tabla de versiones.....</b>	<b>2</b>
<b>Introducción.....</b>	<b>2</b>
<b>Contenido.....</b>	<b>3</b>
<b>Testing de CodeAudit.....</b>	<b>3</b>
Tests positivos y negativos.....	3
create.safe.....	3
delete.safe.....	5
list.safe.....	5
publish.safe.....	5
show.safe.....	6
update.safe.....	6
Tests de hacking.....	6
create.hack.....	6
delete.hack.....	6
list.hack.....	7
publish.hack.....	7
show.hack.....	7
update.hack.....	8
<b>Testing de AuditRecord.....</b>	<b>8</b>
Tests positivos y negativos.....	8
create.safe.....	9
delete.safe.....	10
list.safe.....	11
publish.safe.....	11
show.safe.....	11
update.safe.....	11
Tests de hacking.....	11
create.hack.....	11
delete.hack.....	12
list.hack.....	13
publish.hack.....	13
show.hack.....	14
update.hack.....	14
<b>Pruebas de rendimiento.....</b>	<b>15</b>
<b>Conclusión.....</b>	<b>18</b>
<b>Bibliografía.....</b>	<b>18</b>

# Resumen ejecutivo

Las pruebas de un sistema software son esenciales para garantizar el funcionamiento deseado del producto, permitiendo gran variedad de beneficios como el descubrimiento temprano de bugs, el análisis y comprensión del código, y muchas otras ventajas.

Sin embargo, un testing de calidad implica un procedimiento riguroso comprobando hasta el último recorrido del código a probar mediante una alta cobertura, ya que de lo contrario es posible que queden fallos ocultos a la vista del desarrollador.

## Tabla de versiones

Revisión	Fecha	Descripción
1.0	27-05-2024	Primera versión del documento
2.0	03-07-2024	Revisión para la segunda convocatoria
3.0	15-10-2024	Revisión para la tercera convocatoria

## Introducción

En este documento se ofrece una visión detallada del proceso de elaboración del conjunto de pruebas creadas por el estudiante 5 para las diferentes funcionalidades y características que ofrece el sistema al rol cliente respecto a las entidades CodeAudit y AuditRecord.

Además se incluye un análisis de las pruebas de rendimiento en diferentes ordenadores para comprobar el desempeño del sistema en dichas pruebas.
















### NOTA:

Para el testeo de los campos de tipo @Url y @Date, se tendrán en cuenta lo que hemos considerado límites impuestos por la asignatura a partir del Excel Sample-Data.xlsx así com palabras del profesor Rafael Corchuelo, aunque no siempre se den en el caso práctico, esto es:

- Para enlaces, no se incluirá *ftp://* como caso mínimo, ya que supuestamente está mal formado.
- Para fechas:
  - 01/01/2000 0:00:00 se considerará el momento mínimo y no se introducirán fechas anteriores.
  - 30/07/2022 0:00:00 se considerará el momento presente y no se utilizará en campos con la restricción @Past.

# Contenido

En primer lugar se muestra la cobertura obtenida al reproducir todas las pruebas, puede observarse que todos los servicios superan el 80% sin problemas. De hecho, las líneas marcadas como “amarillas” no muestran ningún tipo de problema ya que la mayoría son “assert object != null;” y algunos cálculos del status para la comprobación de autorización que irremediamente muestran esta cobertura.

▼	acme.features.auditor.codeAudit		94,5 %
>	AuditorCodeAuditUpdateService.java		93,4 %
>	AuditorCodeAuditShowService.java		97,7 %
>	AuditorCodeAuditPublishService.java		94,7 %
>	AuditorCodeAuditListOwnService.java		95,6 %
>	AuditorCodeAuditDeleteService.java		92,6 %
>	AuditorCodeAuditCreateService.java		93,3 %
>	AuditorCodeAuditController.java		100,0 %
▼	acme.features.auditor.auditRecord		93,1 %
>	AuditorAuditRecordUpdateService.java		92,8 %
>	AuditorAuditRecordShowService.java		96,7 %
>	AuditorAuditRecordPublishService.java		89,9 %
>	AuditorAuditRecordListService.java		93,8 %
>	AuditorAuditRecordDeleteService.java		89,5 %
>	AuditorAuditRecordCreateService.java		94,2 %
>	AuditorAuditRecordController.java		100,0 %

## Testing de CodeAudit

### Tests positivos y negativos

Todos los tests positivos y negativos se han llevado a cabo desde la cuenta de **auditor11**.

#### create.safe

En primer lugar se crea un CodeAudit vacío para que resulte en error en los campos no opcionales.

A continuación se completa con valores correctos y se prueban valores negativos en cada campo, atributo por atributo:

- **code**
  - Códigos que no cumplen con el formato:

- *aaa-001*
  - *AAA 001*
  - *-001*
  - *AAAA-001*
  - *AAA001*
  - *codigo*
- Código duplicado:
  - *AAA-000*
- **type**
  - Es un desplegable, no hay margen para tests negativos
- **corrective actions**
  - Texto con spam:
    - *sex sex sex*
  - Texto por encima de la longitud máxima (justo al límite y mucho mayor):
    - *Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore\*\**
    - *Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut laboreLorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore*
- **link**
  - Enlace con longitud superior al límite:
    - *http://www.lorem-ipsum.org/dolor/sit/amet,/consectetur/adipiscing/elit,/sed/do/eiusmod/tempor/incidunt/ut/labore/et/dolore/magna/aliqua/Ut/enim/ad/minim/veniam/quis/nostrud/exercitation/ullamco/laboris/nisi/ut/aliquip/ex/ea/commodo/consequat/duis/aute/XY*
  - Enlace mal formado:
    - *enlace*
- **project**
  - Es un desplegable, no hay margen para tests negativos

A continuación se crea un CodeAudit con valores intermedios (*CRM-000*, *STATIC*, *Lorem ipsum*, *http://www.lorem-ipsum.org*, *Test title*).

Por último se crean múltiples CodeAudits con variedad de casos positivos de todos los atributos:

- **code**
  - *A-001*
  - *AA-001*
  - *AAA-001*
  - *CRM-0XX* (donde XX va desde 04 hasta 12, aumentando en 1 en cada test)
- **type**
  - Se prueban arbitrariamente los dos valores posibles:
    - *STATIC*
    - *DYNAMIC*
- **corrective actions**
  - *L*
  - *Lo*

- *Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore*
- *Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore\**
- `<h1>!</h1>`
- `' or 'A'='A`
- 국민경제의 발전을

○ العظمى واعتلاء

- **link**

- Vacío, ya que no es obligatorio.
- `ftp://a`
- `ftp://a.b`
- `http://www Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute`
- `http://www Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute`
- `ftp://www Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute`
- `http://www Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute`
- `http://example.org?a=1&b=2`
- `http://example.org?`
- `http://example.org/a/b?a&b`
- `https://www Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute`

- **project**

- Se prueban distintos proyectos que aparecen como opciones en el entregable.

## delete.safe

Se eliminan los CodeAudits TST-038 (sin AuditRecords), TST-039 (con uno), TST-042 (con más de uno, todos los publicados con notas iguales), TST-045 (con más de un AuditRecord publicado, nota calculada por la moda de estos) y TST-046 (con nota calculada por desempate de la moda).

## list.safe

Se muestra la lista de CodeAudits del auditor. Los hay publicados y sin publicar, lo que afecta positivamente a la cobertura.

## publish.safe

Se intenta publicar los siguientes CodeAudits:

- **TST-038:** Resulta en error por no tener ningún AuditRecord.
- **TST-039:** Resulta en error por no llegar a la nota mínima de C.
- **TST-042:** Resulta en error por tener AuditRecords no publicados.
- **TST-040:** Se publica exitosamente.

- **TST-045:** Se publica exitosamente, nota calculada por la moda.
- **TST-046:** Se publica exitosamente, nota calculada por desempate de la moda.

## show.safe

Se muestran los CodeAudits TST-038, TST-040, TST-043 y TST-044.

## update.safe

Se actualiza el CodeAudit TST-040 sin modificar, y luego con las mismas variaciones que en el caso **create.safe**, con excepción del atributo **project**, que aparece en este caso como texto y no como opción para editar.

# Tests de hacking

En estos tests participan 2 usuarios: **sin iniciar sesión** (un usuario *any*) y de nuevo **auditor11**.

## create.hack

Crear un CodeAudit solo debe funcionar correctamente a un auditor.

- **Sin iniciar sesión**
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/create>: Fallo por no ser auditor.
- **auditor11**
  - Crear el CodeAudit HCK-001 e inspeccionar el elemento para cambiar:
    - executionDate: 2021/07/30 00:00
    - mark: A
    - draftMode: false
 Al hacer *show* del elemento veremos que los elementos modificados seguirán como antes de intentar cambiarlos.
  - Crear el CodeAudit HCK-002 e inspeccionar el elemento para cambiar el id al de uno existente (375). Al hacer show y volver a inspeccionar podemos ver que se le ha asignado el id que le tocaba y no el elegido.

## delete.hack

Eliminar un CodeAudit solo debe funcionar correctamente a un auditor sobre un CodeAudit propio sin publicar.

- **Sin iniciar sesión**
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/delete?id=333>: Fallo por no ser auditor, CodeAudit ya publicado.
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/delete?id=334>: Fallo por no ser auditor, CodeAudit sin publicar.
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/delete?id=47000000>: Fallo por no ser auditor, CodeAudit inexistente.
- **auditor11**
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/delete?id=333>: Fallo por intentar eliminar un CodeAudit ya publicado de otro editor.

- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/delete?id=334>: Fallo por intentar eliminar un CodeAudit no publicado de otro editor.
- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/delete?id=47000000>: Fallo por intentar eliminar un CodeAudit inexistente.
- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/delete?id=375>: Fallo por intentar eliminar un CodeAudit propio pero ya publicado.

## list.hack

Listar los CodeAudits solo debe funcionar correctamente a un auditor.

- Sin iniciar sesión
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/list>: Fallo por no ser auditor.

## publish.hack

Publicar un CodeAudit solo debe funcionar correctamente a un auditor sobre un CodeAudit propio sin publicar.

- Sin iniciar sesión
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/publish?id=333>: Fallo por no ser auditor, CodeAudit ya publicado.
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/publish?id=334>: Fallo por no ser auditor, CodeAudit sin publicar.
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/publish?id=47000000>: Fallo por no ser auditor, CodeAudit inexistente.
- auditor11
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/publish?id=333>: Fallo por intentar publicar un CodeAudit ya publicado de otro editor.
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/publish?id=334>: Fallo por intentar publicar un CodeAudit no publicado de otro editor.
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/publish?id=47000000>: Fallo por intentar publicar un CodeAudit inexistente.
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/publish?id=375>: Fallo por intentar publicar un CodeAudit propio pero ya publicado.

## show.hack

Mostrar un CodeAudit solo debe funcionar correctamente a un auditor sobre un CodeAudit propio.

- Sin iniciar sesión
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/show?id=333>: Fallo por no ser auditor, CodeAudit ya publicado.
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/show?id=334>: Fallo por no ser auditor, CodeAudit sin publicar.
  - <http://localhost:8082/Acme-SF-D04/auditor/code-audit/show?id=47000000>: Fallo por no ser auditor, CodeAudit inexistente.
- auditor11



- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/show?id=333>: Fallo por intentar mostrar un CodeAudit ya publicado de otro editor.
- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/show?id=334>: Fallo por intentar mostrar un CodeAudit no publicado de otro editor.
- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/show?id=47000000>: Fallo por intentar mostrar un CodeAudit inexistente.

## update.hack

Actualizar un CodeAudit solo debe funcionar correctamente a un auditor sobre un CodeAudit propio sin publicar.

- Sin iniciar sesión

- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/update?id=333>: Fallo por no ser auditor, CodeAudit ya publicado.
- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/update?id=334>: Fallo por no ser auditor, CodeAudit sin publicar.
- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/update?id=47000000>: Fallo por no ser auditor, CodeAudit inexistente.

- auditor11

- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/update?id=333>: Fallo por intentar actualizar un CodeAudit ya publicado de otro editor.
- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/update?id=334>: Fallo por intentar actualizar un CodeAudit no publicado de otro editor.
- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/update?id=47000000>: Fallo por intentar actualizar un CodeAudit inexistente.
- <http://localhost:8082/Acme-SF-D04/auditor/code-audit/update?id=375>: Fallo por intentar actualizar un CodeAudit propio pero ya publicado.
- Acceder al CodeAudit TST-045 e inspeccionar el elemento para cambiar:

- executionDate: 2020/02/15
- draftMode: false
- project: a

Al hacer show veremos que los elementos que hemos modificado seguirán igual que antes de intentar cambiarlos.

- Acceder al CodeAudit TST-046 e inspeccionar el elemento para cambiar:

- correctiveActions: Hack
- id: 371

Al hacer show e inspeccionar veremos que TST-046 mantiene su id original y es el que ha sido editado con el texto "Hack". Al acceder a <http://localhost:8082/Acme-SF-D04/auditor/code-audit/show?id=371> veremos que el CodeAudit con id 371 no ha sido modificado.

# Testing de AuditRecord

## Tests positivos y negativos

Todos los tests positivos y negativos se han llevado a cabo desde la cuenta de **auditor11**.

## create.safe

Se accede a la lista de AuditRecords del Code Audit TST-042 y se crea un AuditRecord vacío para que resulte en error en los campos no opcionales.

A continuación se completa con valores correctos y se prueban valores negativos en cada campo, atributo por atributo:

- **code**
  - Códigos que no cumplen con el formato:
    - *au-1234-567*
    - *AU 1234 567*
    - *AU1234567*
    - *codigo*
    - *AB-1234-567*
  - Código duplicado:
    - *AU-0000-000*
- **periodStart / periodEnd** (Son dos atributos distintos pero trabajaremos con ambos juntos ya que están relacionados por el intervalo que forman):
  - Intervalo no completamente en el pasado:
    - *2020/02/28 16:00*      *2022/07/30 00:01*
    - *2020/02/28 16:00*      *2025/07/30 00:00*
    - *2023/02/28 16:00*      *2025/07/30 00:00*
  - Intervalo menor a una hora:
    - *2020/02/28 16:00*      *2020/02/28 16:59*
    - *2020/02/28 16:00*      *2020/02/28 16:58*
    - *2020/02/28 16:01*      *2020/02/28 17:00*
    - *2020/02/28 16:02*      *2020/02/28 17:00*
  - Inicio antes que fin:
    - *2020/02/28 16:00*      *2020/02/28 15:00*
  - Inicio antes que fin, final en el futuro (prevalece este último error al mostrar el mensaje):
    - *2025/02/28 16:00*      *2020/02/28 15:00*
- **mark**
  - Es un desplegable, no hay margen para tests negativos
- **link**
  - Enlace con longitud superior al límite:
    - *http://www.loremipsum.org/dolor/sit/amet,/consectetur/adipiscing/elit,/sed/do/eiusmod/tempor/incididunt/ut/labore/et/dolore/magna/aliqua/Ut/enim/ad/minim/veniam/quis/nostrud/exercitation/ullamco/laboris/nisi/ut/aliquip/ex/ea/commodo/consequat/duis/aute/XY*
  - Enlace mal formado:
    - *enlace*

A continuación se crea un AuditRecord con valores intermedios (*AU-1234-000*, *2020/02/28 16:00*, *2020/02/28 19:00*, *http://www.lorem-ipsu.org*).

Por último se crean múltiples AuditRecords con variedad de casos positivos de todos los atributos:

- **code**

- AU-1234-0XX (donde XX es de 00 a 12, aumentando por cada test)
- **periodStart / periodEnd** (Son dos atributos distintos pero trabajaremos con ambos juntos ya que están relacionados por el intervalo que forman):
  - 2020/02/28 16:00      2020/02/28 19:00
  - 2020/02/28 16:00      2022/07/29 23:59
  - 2020/02/28 16:00      2022/07/29 23:58
  - 2000/01/01 00:00      2020/02/28 19:00
  - 2000/01/01 00:01      2020/02/28 19:00
  - 2000/01/01 00:00      2022/07/29 23:59
  - 2000/01/01 00:01      2022/07/29 23:59
  - 2000/01/01 00:00      2022/07/29 23:58
  - 2020/02/28 16:00      2020/02/28 17:00
  - 2020/02/28 16:00      2020/02/28 17:01
  - 2020/02/28 15:59      2020/02/28 17:00
- **mark**
  - Se prueban los distintos valores posibles a seleccionar en el desplegable:
    - APlus
    - A
    - B
    - C
    - F
    - FMinus
    - None
- **link**
  - Vacío, ya que no es obligatorio.
  - ftp://a
  - ftp://a.b
  - http://www.lorem-ipsum.org/dolor/sit/amet,/consectetur/adipiscing/elit,/sed/do/eiusmod/tempor/incidunt/ut/labore/et/dolore/magna/aliqua/Ut/enim/ad/minim/veniam/quis/nostrud/exercitation/ullamco/laboris/nisi/ut/aliquip/ex/ea/commodo/consequat/duis/aute/
  - http://www.lorem-ipsum.org/dolor/sit/amet,/consectetur/adipiscing/elit,/sed/do/eiusmod/tempor/incidunt/ut/labore/et/dolore/magna/aliqua/Ut/enim/ad/minim/veniam/quis/nostrud/exercitation/ullamco/laboris/nisi/ut/aliquip/ex/ea/commodo/consequat/duis/aute/X
  - ftp://www.lorem-ipsum.org
  - http://www.lorem-ipsum.org/dolor/sit.html
  - http://www.lorem-ipsum.org/dolor/sit.html#dolor
  - http://example.org?a=1&b=2
  - http://example.org?
  - http://example.org/a/b?a&b
  - https://www.lorem-ipsum.org

## delete.safe

Se eliminan los AuditRecords AU-7357-47 (del CodeAudit TST-042) y AU-7357-45 (de TST-041).

## list.safe

Se muestran los AuditRecords del CodeAudit TST-038 (sin AuditRecors), de TST-040 (con 1), TST-041 (con 2) y TST-042 (con 3).

## publish.safe

Se intenta publicar los siguientes AuditRecords:

- **AU-7357-47 (de TST-042)**: Resulta en error por tener mark = None.
- **AU-7357-45 (de TST-041)**: Se publica exitosamente.

## show.safe

Se muestran los AuditRecords AU-7357-047 (de TST-042) y AU-7357-045 (de TST-041).

## update.safe

Se actualiza el AuditRecord AU-7357-045 (de TST-041) sin modificar, y luego con las mismas variaciones que en el caso **create.safe**.

# Tests de hacking

En estos tests participan 2 usuarios: **sin iniciar sesión** (un usuario *any*) y de nuevo **auditor11**.

## create.hack

Crear un AuditRecord solo debe funcionar correctamente a un auditor sobre un CodeAudit propio sin publicar.

- **Sin iniciar sesión**
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/create?codeAuditId=333>: Fallo por no ser auditor, CodeAudit ya publicado.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/create?codeAuditId=334>: Fallo por no ser auditor, CodeAudit sin publicar.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/create?codeAuditId=47000000>: Fallo por no ser auditor, CodeAudit inexistente.
- **auditor11**
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/create?codeAuditId=333>: Fallo por intentar crear un AuditRecord en un CodeAudit ya publicado de otro editor.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/create?codeAuditId=334>: Fallo por intentar crear un AuditRecord en un CodeAudit no publicado de otro editor.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/create?codeAuditId=47000000>: Fallo por intentar crear un AuditRecord en un CodeAudit inexistente.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/create?codeAuditId=375>: Fallo por intentar crear un AuditRecord en un CodeAudit propio pero ya publicado.

- Acceder a los AuditRecords del CodeAudit TST-042, crear el AuditRecord AU-9000-001 e inspeccionar el elemento para cambiar:

- `draftMode: false`

Al hacer *show* del elemento veremos que el elemento modificado seguirá como antes de intentar cambiarlo.

- Acceder a los AuditRecords del CodeAudit TST-042, crear el AuditRecord AU-9000-002 e inspeccionar el elemento para cambiar el id al de uno existente (422). Al hacer *show* y volver a inspeccionar podemos ver que se le ha asignado el id que le tocaba y no el elegido, y al acceder a <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=422> podemos ver que el AuditRecord con id 422 está intacto.

## delete.hack

Eliminar un AuditRecord solo debe funcionar correctamente a un auditor sobre un AuditRecord sin publicar hijo de un CodeAudit propio.

- **Sin iniciar sesión**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=381>: Fallo por no ser auditor, AuditRecord ya publicado hijo de un CodeAudit ya publicado.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=387>: Fallo por no ser auditor, Audit Record ya publicado hijo de un CodeAudit sin publicar.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=382>: Fallo por no ser auditor, Audit Record sin publicar.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=47000000>: Fallo por no ser auditor, AuditRecord inexistente.

- **auditor11**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=381>: Fallo por intentar eliminar un AuditRecord ya publicado hijo de un CodeAudit ya publicado de otro auditor.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=387>: Fallo por intentar eliminar un AuditRecord ya publicado hijo de un CodeAudit sin publicar de otro auditor.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=382>: Fallo por intentar eliminar un AuditRecord sin publicar hijo de un CodeAudit sin publicar de otro auditor.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=47000000>: Fallo por intentar eliminar un AuditRecord inexistente.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=427>: Fallo por intentar eliminar un AuditRecord ya publicado hijo de un CodeAudit propio ya publicado.
  - <http://localhost:8082/Acme-SF-D04/auditor/audit-record/delete?id=418>: Fallo por intentar eliminar un AuditRecord ya publicado hijo de un CodeAudit propio sin publicar.

## list.hack

Listar AuditRecords solo debe funcionar correctamente a un auditor sobre un CodeAudit propio.

- **Sin iniciar sesión**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/list?codeAuditId=333>  
: Fallo por no ser auditor, CodeAudit ya publicado.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/list?codeAuditId=334>  
: Fallo por no ser auditor, CodeAudit sin publicar.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/list?codeAuditId=4700000>: Fallo por no ser auditor, CodeAudit inexistente.

- **auditor11**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/list?codeAuditId=333>  
: Fallo por intentar listar los AuditRecords hijos de un CodeAudit ya publicado de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/list?codeAuditId=334>  
: Fallo por intentar listar los AuditRecords hijos de un CodeAudit sin publicar de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/list?codeAuditId=4700000>: Fallo por intentar listar los AuditRecords hijos de un CodeAudit inexistente.

## publish.hack

Publicar un AuditRecord solo debe funcionar correctamente a un auditor sobre un AuditRecord sin publicar hijo de un CodeAudit propio.

- **Sin iniciar sesión**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/publish?id=381>: Fallo por no ser auditor, AuditRecord ya publicado hijo de un CodeAudit ya publicado.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/publish?id=387>: Fallo por no ser auditor, AuditRecord ya publicado hijo de un CodeAudit sin publicar.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/publish?id=382>: Fallo por no ser auditor, AuditRecord sin publicar.

- **auditor11**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/publish?id=381>: Fallo por intentar publicar un AuditRecord ya publicado hijo de un CodeAudit ya publicado de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/publish?id=387>: Fallo por intentar publicar un AuditRecord ya publicado hijo de un CodeAudit sin publicar de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/publish?id=382>: Fallo por intentar publicar un AuditRecord sin publicar hijo de un CodeAudit de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/publish?id=4700000>: Fallo por intentar publicar un AuditRecord inexistente.

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/publish?id=427>: Fallo por intentar publicar un AuditRecord ya publicado hijo de un CodeAudit propio ya publicado.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/publish?id=418>: Fallo por intentar publicar un AuditRecord ya publicado hijo de un CodeAudit propio sin publicar.

## show.hack

Publicar un AuditRecord solo debe funcionar correctamente a un auditor sobre un AuditRecord hijo de un CodeAudit propio.

- **Sin iniciar sesión**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=381>: Fallo por no ser auditor, AuditRecord ya publicado hijo de CodeAudit ya publicado.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=387>: Fallo por no ser auditor, AuditRecord ya publicado hijo de CodeAudit sin publicar.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=382>: Fallo por no ser auditor, AuditRecord sin publicar.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=47000000>: Fallo por no ser auditor, AuditRecord inexistente.

- **auditor11**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=381>: Fallo por intentar mostrar un AuditRecord ya publicado hijo de un CodeAudit ya publicado de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=387>: Fallo por intentar mostrar un AuditRecord ya publicado hijo de un CodeAudit sin publicar de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=382>: Fallo por intentar mostrar un AuditRecord sin publicar hijo de un CodeAudit de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=47000000>: Fallo por intentar mostrar un AuditRecord inexistente.

## update.hack

Actualizar un AuditRecord solo debe funcionar correctamente a un auditor sobre un AuditRecord sin publicar hijo de un CodeAudit propio.

- **Sin iniciar sesión**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=381>: Fallo por no ser auditor, AuditRecord ya publicado hijo de un CodeAudit ya publicado.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=387>: Fallo por no ser auditor, AuditRecord ya publicado hijo de un CodeAudit sin publicar.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=382>: Fallo por no ser auditor, AuditRecord sin publicar.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=47000000>: Fallo por no ser auditor, AuditRecord inexistente.

- **auditor11**

- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=381>: Fallo por intentar actualizar un AuditRecord ya publicado hijo de un CodeAudit ya publicado de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=387>: Fallo por intentar actualizar un AuditRecord ya publicado hijo de un CodeAudit sin publicar de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=382>: Fallo por intentar actualizar un AuditRecord sin publicar hijo de un CodeAudit de otro auditor.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=47000000>: Fallo por intentar actualizar un AuditRecord inexistente.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=427>: Fallo por intentar actualizar un AuditRecord ya publicado hijo de un CodeAudit propio ya publicado.
- <http://localhost:8082/Acme-SF-D04/auditor/audit-record/update?id=418>: Fallo por intentar actualizar un AuditRecord ya publicado hijo de un CodeAudit propio sin publicar.
- Acceder a los AuditRecords del CodeAudit TST-042, de ahí acceder al AuditRecord AU-7357-047 e inspeccionar el elemento para cambiar:
  - `draftMode: false`

Al hacer show, veremos que lo que hemos modificado se mantiene igual que antes de intentar cambiarlo.
- Acceder a los AuditRecords del CodeAudit TST-042, de ahí acceder al AuditRecord AU-7357-047 e inspeccionar el elemento para cambiar:
  - `link: http://hack.com`
  - `id: 421`

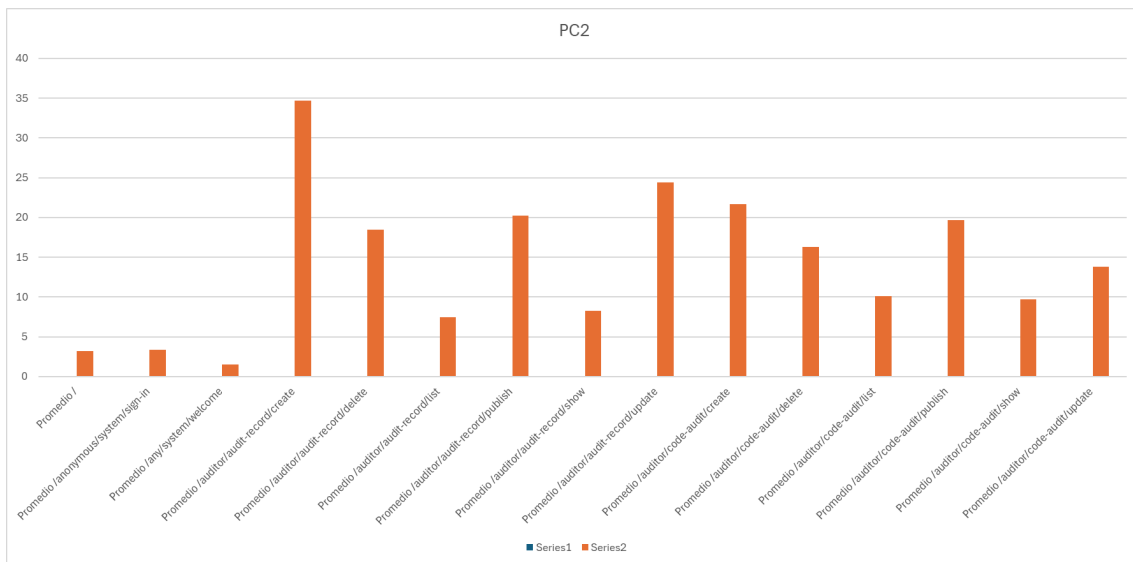
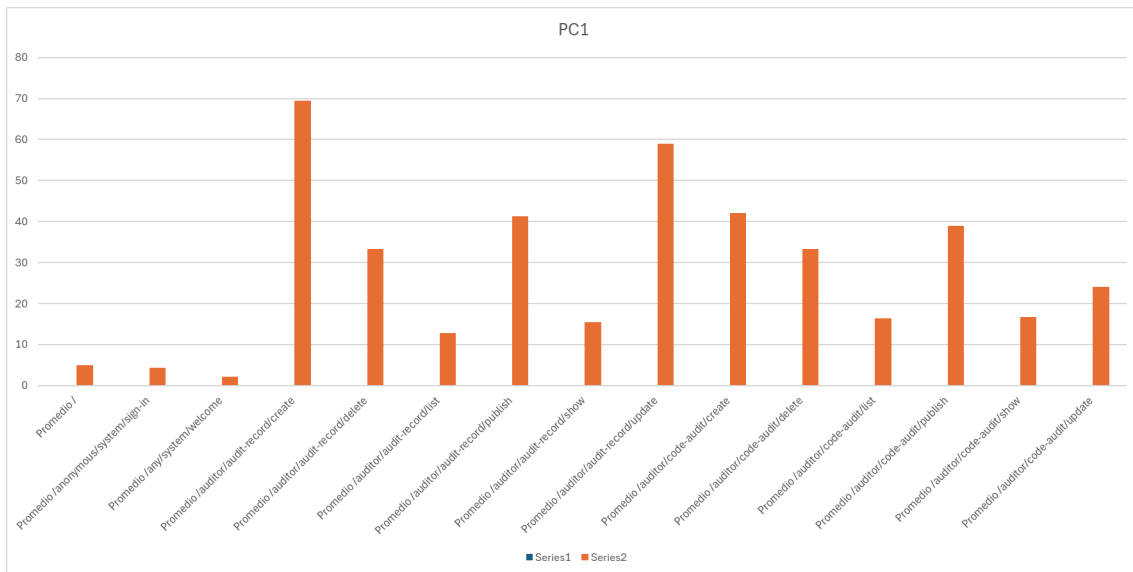
Al hacer show veremos que es AU-7327-047 el que hemos editado, e inspeccionando elemento veremos que mantiene su id original. Accediendo a <http://localhost:8082/Acme-SF-D04/auditor/audit-record/show?id=421> veremos que el AuditRecord con id 421 no ha sido modificado.

## Pruebas de rendimiento

Para probar el rendimiento del sistema respecto a las pruebas se ha lanzado el replayer en 2 ordenadores personales distintos. En ambos casos se han añadido índices a las entidades consultadas en las pruebas grabadas para mejorar el rendimiento, además de haber sido los tests grabados siendo los únicos programas en ejecución en las máquinas para liberar todos los recursos posibles.

Los gráficos de tiempo obtenidos son los siguientes:





A simple vista las gráficas nos muestran una clara ventaja del segundo ordenador, ya que podemos ver que aunque tengan una forma similar, el eje vertical de tiempo del segundo está en una escala menor.

Observemos las estadísticas y los intervalos de confianza para cada caso:

PC1				PC2		
Media	25,0859568			Media	13,0433314	
Error típico	1,26362994			Error típico	0,58661507	
Mediana	11,8883995			Mediana	7,784849	
Moda	#N/D			Moda	19,2654	
Desviación estándar	29,6347489			Desviación estándar	13,7573429	
Varianza de la muestra	878,218342			Varianza de la muestra	189,264485	
Curtosis	8,82135107			Curtosis	7,1120984	
Coeficiente de asimetría	2,34721605			Coeficiente de asimetría	2,21692054	
Rango	256,190699			Rango	104,0177	
Mínimo	1,3238			Mínimo	0,8144	
Máximo	257,514499			Máximo	104,8321	
Suma	13797,2762			Suma	7173,83229	
Cuenta	550			Cuenta	550	
Nivel de confianza(95,0%)	2,48214127			Nivel de confianza(95,0%)	1,15228473	
Intervalo (ms)	22,6038155	27,568098		Intervalo (ms)	11,8910467	14,1956162
Intervalo (s)	0,02260382	0,0275681		Intervalo (s)	0,01189105	0,01419562

De nuevo podemos observar mejores resultados en el segundo, ya que los tiempos son menores en todas las estadísticas que medimos, así como el tamaño del intervalo: (22.6, 27.57) en el primero frente a (11.89, 14.2) en el segundo.

A continuación podemos ver el resultado de la prueba z:

Prueba z para medias de dos muestras		
	PC1	PC2
Media	25,0859568	13,0433314
Varianza (conocida)	878,21834	189,26448
Observaciones	550	550
Diferencia hipotética de las medias	0	
z	8,64414558	
P(Z<=z) una cola	0	
Valor crítico de z (una cola)	1,64485363	
Valor crítico de z (dos colas)	0	
Valor crítico de z (dos colas)	1,95996398	

Estableciendo el contraste de hipótesis, tenemos la hipótesis nula ( $H_0$ ) y la hipótesis alternativa ( $H_1$ ) siguientes:

- $H_0$ : Uno de los ordenadores es más eficiente que el otro y podemos establecer una comparación.
- $H_1$ : Ambos ordenadores tienen una eficiencia personal, por lo que no podemos compararlos.

Para un nivel de confianza del 95%, tenemos un nivel de significación ( $\alpha$ ) de 0.05. El p-value obtenido es 0, y como se encuentra en el intervalo  $[0.00, \alpha)$  podemos aceptar la hipótesis nula: uno de los ordenadores es, en efecto, más eficiente que el otro.

Para ver cuál de los ordenadores es más eficiente, comparamos las medias de tiempo de ejecución en cada uno:

PC1	PC2	Diferencia
25,0859568 ms	13,0433314 ms	12,0426253 ms

Podemos comprobar que, tal como dedujimos anteriormente, el segundo ordenador es mucho más eficiente por una diferencia de más de 12 milisegundos de media.

## Conclusión

Se ha realizado un testeo exhaustivo de las funcionalidades, aunque ha requerido mucho más tiempo del esperado debido al carácter manual del procedimiento, así como a su fragilidad (un cambio en la implementación de una funcionalidad fácilmente requiere una nueva grabación de tests). Con todo, podemos considerar la variedad de tests satisfactoria por su amplio rango de cobertura.

Respecto al estudio estadístico, podemos ver un resultado relativamente lento en el primer ordenador, y llegar a una clara conclusión de que las prestaciones del equipo influyen en la reproducción de los tests y la eficiencia al llevarla a cabo.

## Bibliografía

No aplica.