Universidad de Sevilla  
Escuela Técnica Superior de Ingeniería Informática

TESTING REPORT - D04



**C1.04.02**

**Repositorio**: [https://github.com/josrojrom1/C1.04.02](https://github.com/josrojrom1/C1.04.02%C3%A7)

**Autor:**

Ybarra Manrique, Miguel, migybaman[@alum.us.es](mailto:josrojrom1@alum.us.es)

**Fecha:** 23/05/2023

**Tabla de contenidos**

[**Resumen**](#_4hmpm5uc85lj) **2**

[**Tabla de revisión**](#_90otg4vxm3ka) **2**

[**Introducción**](#_58vtmc9pw6vo) **2**

[**Contenido**](#_tvy5edcq3b5t) **3**

[**Bibliografía**](#_ysgwrb1dc6k3) **11**

**Resumen**

En este documento se va a llevar a cabo un desglose detallado a cerca del

*testing suite* completo obtenido por el *coverage* de las pruebas unitarias *JUnit* realizadas sobre el proyecto en su versión final, en concreto sobre las funcionalidades relacionadas con *Peep*.

**Tabla de revisión**

| **Versión** | **Fecha** | **Descripción** |
| --- | --- | --- |
| V1 | 25/05/2023 | Creación del testing report |

**Introducción**

El requisito a evaluar por los tests eran el #14 y #15 que dicen lo siguiente:

***#14 Operations by assistants on tutorials:***

***-List the tutorials that they have created.***

***-Show the details of their tutorials.***

***-Create, update, or delete their tutorials, as long as they have not been published.***

***#15 Operations by assistants on sessions:***

***-List the sessions in their tutorials.***

***-Show the details of their sessions.***

***-Create a new session in a tutorial, as long as it has been published.***

***-Update or delete the sessions in their tutorials, as long as they have not been published.***

**Testing funcional**

**Funcionalidad:** *Tutorial*

**Rol:** *Assistant*

**Test:** List Tutorial with Rol Assistant

**Ruta:** *“acme>testing>assistant>tutorial>AssistantTutorialListTest.java”*

Para el **list** de *Tutorials* se desarrollaron dos casos distintos:

* En el primer caso era uno positivo, donde una vez iniciado sesión como *assistant1,* se accede a al listado de las tutorías de dicho usuario y a través de los datos en el csv correspondiente, se comprueba que los datos que muestra son los correctos.
* En el segundo caso eran los tests de hacking, donde comprueba que un usuario no autenticado o un usuario que no tenga el rol de asistente no pueda acceder a la información. Se probó con anónimo y con el usuario *lecturer1*.

Con estos casos de testing para el listado no se han detectado ningún tipo de bugs.

**Test:** Show Tutorial with Rol Assistant

**Ruta:** *“acme>testing>assistant>tutorial>AssistantTutorialShowTest.java”*

Para el test ***show***de *tutorial* se desarrolló los siguientes casos:

* Para comprobar que la información del *Tutorial* se mostrase adecuadamente se creó el fichero *.csv* con un caso de *show* positivo. Para llevarlo a cabo de manera correcta, una vez que se accede al listado, este se ordena con orden ascendente en el campo *code*, así nos aseguramos de que el primer elemento de la lista sea el que intentamos comprobar. Una vez ordenado se accede y se comprueba que los campos tienen los datos correctos.
* Para asegurarnos que la información de un *tutorial* no sea visible para cualquier persona creamos el test de hacking. Para ellos hago una query a la base de datos donde extraigo todas la *tutorials* del *assistant1,* después con un bucle for itero la colección y creo un parámetro donde le paso el id de las tutorías y hago la request con distintos principales. Se comprueba con el usuario anónimo, el *assistant2* y con *lecturer1.*  Ninguno de ellos debería tener acceso a las tutorías por tanto aparece una pantalla de pánico mostrando que no están autorizados.

Con estos casos de testing para el listado no se han detectado ningún tipo de bugs

**Test:** Create Tutorial with Rol Assistant

**Ruta:** *“acme>testing>assistant>tutorial>AssistantTutorialCreateTest.java”*

Para el **create** de *Tutorials* se desarrollaron los siguientes casos:

* El primer caso es el test positivo de *create tutorial* donde se siguen los siguientes pasos logueado como *assistant1*: Click en *“Assistants > List my tutorials”* y de nuevo click en *Create a tutorial*, después se introducen todos los campos del formulario con las entradas creadas en el fichero *.csv*. Una vez rellenos se hace click en el botón de create y al final se comprueba que no existen errores al publicar el Tutorial y que se ha creado correctamente y aparecen todos los campos.
* El segundo caso es el test negativo. En este test se repiten los mismos pasos que en el test anterior pero esta vez en el fichero *.csv* se introducen entradas erróneas comprobando que efectivamente se activan las validaciones correspondientes y que en ningún caso podemos crear un *Tutorial*. Se hacen comprobaciones de entradas en blanco, datos erróneos, o spam. para detectar que ocurran los problemas de manera esperada, en cada entrada del csv se pone un solo error.
* El tercer caso es el test de hacking. Donde se comprueba que ningún principal que no tenga el rol de asistente tenga la autorización de acceder al formulario ni crear un tutorial.

Con estos tests comprobé que no hice de manera correcta el *authorise* y al entrar mostraba un error en el getAreaWorkData, porque no encontraba el id del supuesto asistente que entraba, esta comprobación la hacía porque de la manera que está hecha hace que ponga directamente en el objeto el asistente logueado. Por tanto lo edité para que saltase el NotAuthorised.

**Test:** Update Tutorial with Rol Assistant

**Ruta:** *“acme>testing>assistant>tutorial>AssistantTutorialUpdateTest.java”*

Para el **update** de *Tutorials* se desarrollaron los siguientes casos:

* El primer caso es el test positivo de *update* donde una vez logueado como *assistant2* se siguen los siguientes pasos, se hace click en el menú Assistant y después a List my tutorials, ordena la lista de manera ascendente por el código, para asegurarnos que escogemos el correcto. Una vez ordenado entramos al primer tutorial y rellenamos los campos con los datos que aparecen en el csv, se clicka el botón de update y se comprueba que los campos han sido modificados correctamente.
* El segundo caso es el test negativo. En este test se repiten los mismos pasos que en el test anterior pero esta vez en el fichero *.csv* se introducen entradas erróneas comprobando que efectivamente se activan las validaciones correspondientes y que en ningún caso podemos modificar un *Tutorial*. Se hacen comprobaciones de entradas en blanco, datos erróneos, o spam. para detectar que ocurran los problemas de manera esperada, en cada entrada del csv se pone un solo error.
* El tercer caso es el test de hacking. Donde se comprueba que ningún principal que no tenga el rol de asistente tenga la autorización de acceder al formulario ni crear un tutorial.

Con estos test no encontré ningún bug significativo, simplemente encontré incongruencias en los archivos csv iniciales.

**Test:** Publish Tutorial with Rol Assistant

**Ruta:** *“acme>testing>assistant>tutorial>AssistantTutorialPublishTest.java”*

Para el **Publish** de *Tutorials* se desarrollaron los siguientes casos:

* El primer caso es el test positivo de *publish* donde una vez logueado como *assistant2* se siguen los siguientes pasos, se hace click en el menú Assistant y después a List my tutorials, ordena la lista de manera ascendente por el código, para asegurarnos que escogemos el correcto. Una vez ordenado entramos al primer tutorial y clickamos en el botón publish. Después se comprueba que verdaderamente se haya publicado el tutorial, para ello se comprueba que no existan los botones de update, delete y publish.
* El segundo caso es el test de hacking, donde se comprueba que los usuarios que no sean el creador del tutorial puedan publicarlo. Para ello se prueba sacando todos los tutorials de assistant2 y con un bucle for los itero y compruebo individualmente, con diferentes usuarios.

Con estos tests no se encontró ningún bug.

**Funcionalidad:** *TutorialSession*

**Rol:** *Assistant*

**Test:** Create TutorialSession with Rol Assistant

**Ruta:** *“acme>testing>assistant>tutorialSession>AssistantTutorialSessionCreateTest.java”*

Para el **create** de *TutorialSession* se desarrollaron los siguientes casos:

* El primer caso es el test positivo de *create tutorialSession* donde se siguen los siguientes pasos logueado como *assistant2*: Click en *“Assistant > List my tutorials”* y de nuevo click en el primer tutorial, ahora se hace click en *List session* y seguidamente a *create a sessions. En el formulario que aparece* se introducen todos los campos del formulario con las entradas creadas en el fichero *.csv*. Una vez rellenos se hace click en el botón submit de create y al final se comprueba que no existen errores al crear el TutorialSession y que se ha creado correctamente y aparecen todos los campos.
* El segundo caso es el test negativo. En este test se repiten los mismos pasos que en el test anterior pero esta vez en el fichero *.csv* se introducen entradas erróneas comprobando que efectivamente se activan las validaciones correspondientes y que en ningún caso podemos crear un *TutorialSession*. Se hacen comprobaciones de entradas en blanco, datos erróneos, o spam. Para detectar que ocurran los problemas de manera esperada, en cada entrada del csv se pone un solo error.
* El tercer caso es el test de hacking. Donde se comprueba que ningún principal que no tenga el rol de asistente tenga la autorización de acceder al formulario ni crear un tutorialSession. También se comprueba que no se pueda crear si el tutorial al que está relacionado ya está publicado.

Con estos tests no se encontró ningún bug.

**Test:** Update TutorialSession with Rol Assistant

**Ruta:** *“acme>testing>assistant>tutorialSession>AssistantTutorialSessionUpdateTest.java”*

Para el **Update** de *TutorialSession* se desarrollaron los siguientes casos:

* El primer caso es el test positivo de *update tutorialSession* donde se siguen los siguientes pasos logueado como *assistant2*: Click en *“Assistant > List my sessions”*, ahora se hace click en el primer *session. En el formulario que aparece* se introducen todos los campos del formulario con las entradas creadas en el fichero *.csv*. Una vez rellenos se hace click en el botón submit de update y al final se comprueba que no existen errores al modificar el TutorialSession y que se ha modificado correctamente y aparecen todos los campos.
* El segundo caso es el test negativo. En este test se repiten los mismos pasos que en el test anterior pero esta vez en el fichero *.csv* se introducen entradas erróneas comprobando que efectivamente se activan las validaciones correspondientes y que en ningún caso podemos modificar un *TutorialSession*. Se hacen comprobaciones de entradas en blanco, datos erróneos, o spam. Para detectar que ocurran los problemas de manera esperada, en cada entrada del csv se pone un solo error.
* El tercer caso es el test de hacking. Donde se comprueba que ningún principal que no tenga el rol de asistente tenga la autorización de acceder al formulario ni modificar un tutorialSession. También se comprueba que no se pueda modificar si el tutorial al que está relacionado ya está publicado.

Con estos tests no se encontró ningún bug.

**Test:** List TutorialSession with Rol Assistant

**Ruta:** *“acme>testing>assistant>tutorialSession>AssistantTutorialSessionListTest.java”*

Para el **list** de *TutorialSession* se desarrollaron tres casos distintos:

* En el primer caso era uno positivo, donde una vez iniciado sesión como *assistant2,* se hace click en el menú assistant y se le da al botón List my sessions, una vez dentro se comprueba que los datos sean correctos.
* En el segundo caso es también uno positivo, donde una vez iniciado sesión como *assistant2,* se hace click en el menú assistant y se le da al botón List my tutorials, la lista de tutorías se ordena, se accede al primero y se clicka en el botón de *list sessions*, una vez dentro se comprueba que los datos sean correctos.
* En el tercer caso eran los tests de hacking, donde comprueba que un usuario no autenticado o un usuario que no tenga el rol de asistente no pueda acceder a la información. Se probó con anónimo, con el usuario *lecturer1*  y con el assistant1.

Con estos casos de testing para el listado no se han detectado ningún tipo de bugs.

**Test:** Show TutorialSession with Rol Assistant

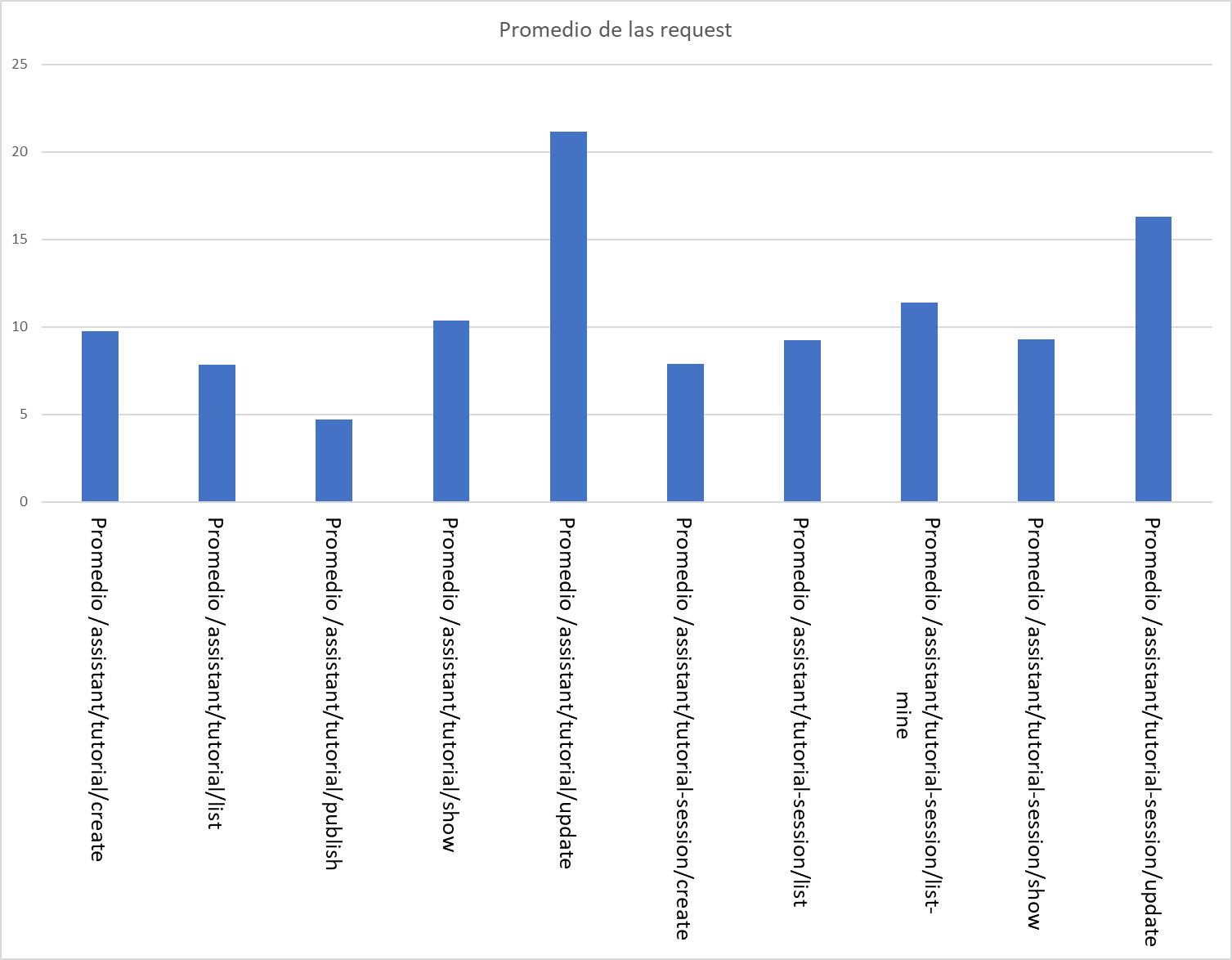
**Ruta:** *“acme>testing>assistant>tutorial>AssistantTutorialShowTest.java”*

Para el test ***show***de *tutorial* se desarrolló los siguientes casos:

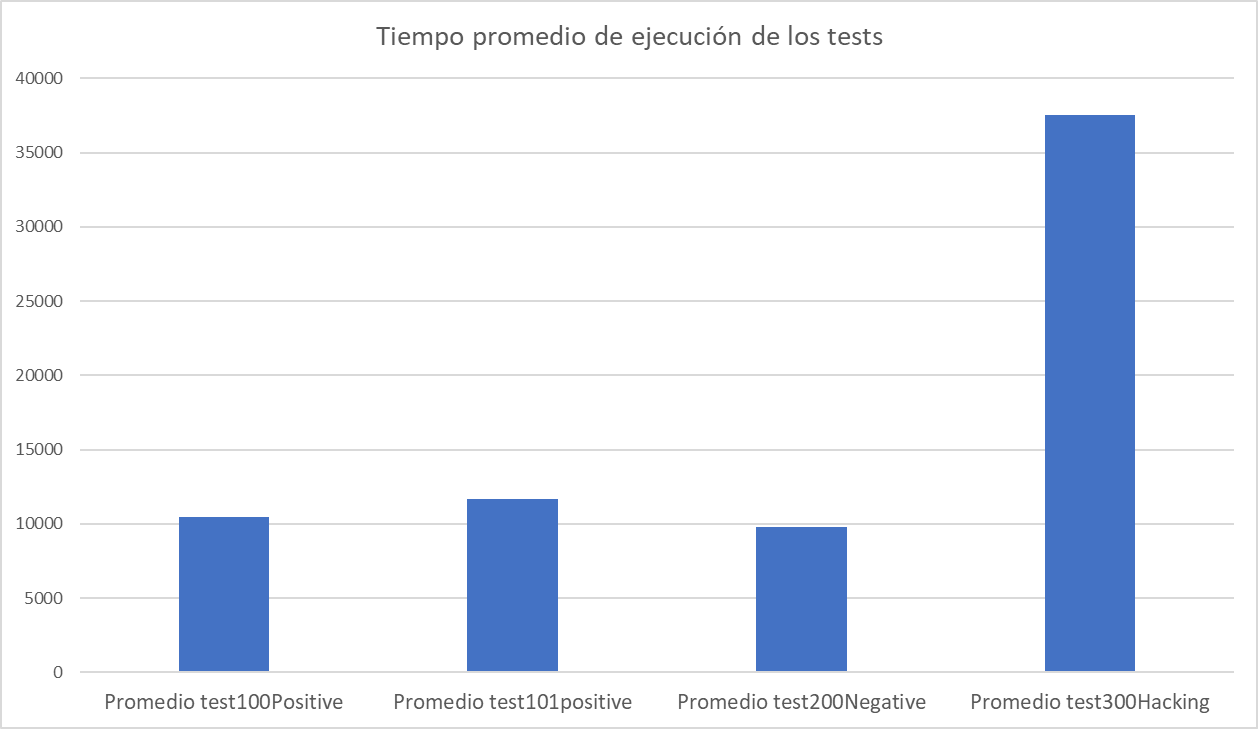
* En el primer caso, inicia sesión como assistant2 y hace click en el botón del menú Assistant y seguidamente al de List my sessions, en esta lista entra al primero y comprueba que la información se muestra correctamente.
* En el segundo caso, se inicia otra vez con assistant2, pero esta vez se accede a través de una tutoría en específico, para ellos clicka en el el menú en el botón de assistant->List my tutorials. En la lista de tutorías se accede a la primera y dentro se clicka al botón de list sessions, una vez dentro se vuelve a clickar en el primer acceso de la lista y se comprueban todos lo valores.
* En el tercer caso eran los tests de hacking, donde comprueba que un usuario no autenticado o un usuario que no tenga el rol de asistente no pueda acceder a la información. Se probó con anónimo, con el usuario *lecturer1*  y con el assistant1.

Para estos test ocurrían lo que pensaba que era un bug, porque no me detectaba correctamente los valores de dos select. Una vez preguntado en un follow up, la profesora me indicó que para hacer la búsqueda en la vista debía poner “\_proxy” después del nombre del atributo que buscaba para que se hiciese correctamente. A parte de este apunte, no encontré ningún bug.

**Testing de rendimiento**



[ Eje Y en milisegundos | Eje X promedio de cada test por funcionalidad/entidad/rol ]



[ Eje Y en milisegundos | Eje X promedio de cada test individual ]

Usando la herramienta excel, tal y como viene explicado en la teoría, para analizar los datos, podemos decir que el intervalo en milisegundos del nivel de confianza al 95% es de 9,49ms.

Esta evaluación del rendimiento de los tests se hizo ejecutando los tests en una máquina con un intel i7 de décima generación con 16GB de memoria RAM y un disco SSD.

# Bibliografía

* Documento 08 Annexes, de los contenidos de la plataforma virtual de esta asignatura.
* Documentos del lesson L04 - formal testing