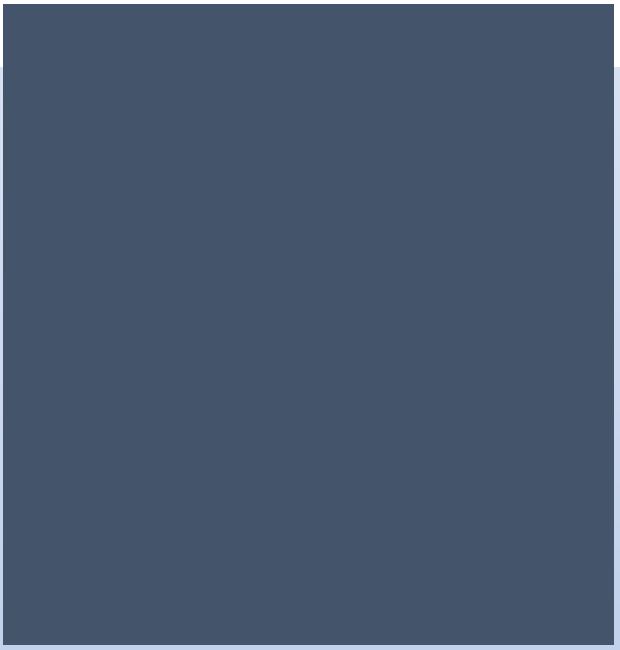


Título

Diseño y Pruebas 2

C1.03.08

Contenido

1. Tabla de versiones

Tabla de versiones		
Versión	**Fecha**	**Descripción**
1.0	25/05/2023	D04

2. Resumen ejecutivo

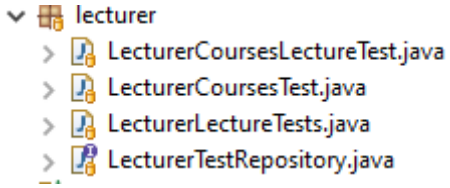
Este documento contiene el análisis de los casos de tests referentes al entregable D04 correspondiente al "Student 1" correspondiente a Juan José Casamitjana Benítez.

3. Introducción

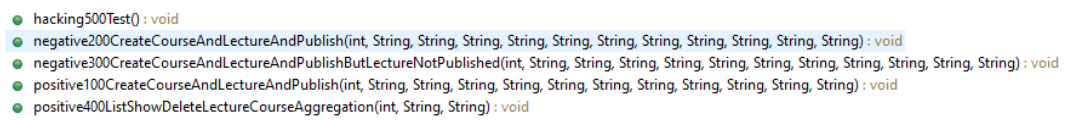
Aquí se van a analizar los casos de test que se han tenido en cuenta, como se han implementado y como ha sido su desempeño

4. Contenidos

- Tests funcionales: Por lo general estos tests han sido útiles a la hora de encontrar errores generales en todas las funcionalidades correspondientes a los Lecturer, además han jugado un papel importante a la hora de comprobar que la aplicación se estaba integrando y ejecutando correctamente.



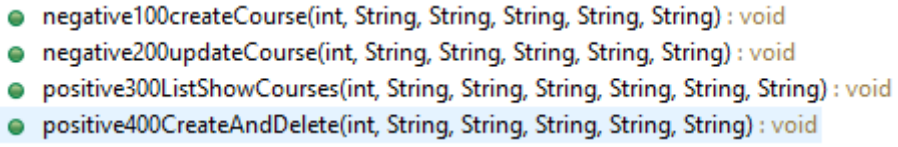
  - LecturerCoursesLectureTest: Esta clase se encarga de testear las funcionalidades de la relación intermedia entre Course y Lecture que se definió en otros entregables



hacking500Test: Se obtienen las relaciones curso lección del lecturer1 e iniciamos sesión con el lecturer2, posteriormente intentamos acceder por url a cada una de ellas por cada uno de los comandos posibles.

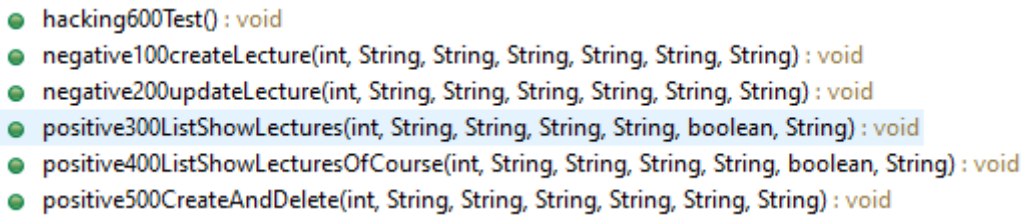
negative200CreateCourseAndLectureAndPublish: Este test negativo crea un curso y una lecture de tipo teorica y la intenta publicar con el objetivo de obtener el error de que no se pueden publicar cursos puramenete teoricos.
negative300CreateCourseAndLectureAndPublishLectureNotPublished: En este test se hace lo mismo que en anterior, la diferencia es que la Lecture creada es “hands on” pero no es publicada despues de ser creada, al intentar publicar el curso se busca que salte el error correspondiente a esta regla de negocio.
positive100CreateCourseAndLectureAndPublish: Este test hace lo mismo que los anteriores pero esta vez buscamos que todo vaya correctamente, luego la lección es publicada y es no teorica.
positive400ListShowDeleteLectureCourseAggregation: Se sigue el siguiente flujo, se accede al listado de relaciones curso lección del lecturer1 se comprueban que son correctas tanto en el listado como en el show y se borran.

- LecturerCourses: Estos tests están orientados a casos de uso generalmente relacionados con los cursos.



negative100createCourse: Este test está exclusivamente hecho para probar los casos negativos de la creación de cursos.
negative200updateCourse: Funciona igual que el anterior pero ahora validamos que también se cumplan las restricciones en el update.
positive300ListShowCourses: Iniciamos sesión con el lecturer1 y comprobamos que todos los datos de sus cursos son correctos.
positive400CreateAndDelete: Iniciamos sesión con un nuevo lecturer y creamos cursos y los borramos de forma repetida.
hacking500Test: Se obtienen los cursos del lecturer1 e iniciamos sesión con el lecturer2, luego se intenta acceder a cada uno de los comandos que permitan editar un curso, además se intenta acceder al listado de lecciones de ese curso.

- LecturerLecture: Estos tests prueban todas las funcionalidades relacionadas con Lecture y sus casos de uso



negative100createLecture: Casos en los que nos da error el formulario de creación de Lecture.
negative200updateLecture: Casos en los que nos da error el formulario de actualización de Lecture.
positive300ListShowLectures: Iniciamos sesión con el lecturer1 y comprobamos que todas sus lecciones se presenten correctamente.
positive400ListShowLecturesOfCourse: Se comprueba que se muestran correctamente las lecciones de un curso que hayamos creado.

positive500CreateAndDelete: Se crean y borran lecciones de forma repetida para ver que todo funciona correctamente.
hacking600Test: Se obtienen las lecciones del lecturer1 e iniciamos sesión con el lecturer2, luego intentamos acceder a cada uno de los comandos para intentar modificar o ver los datos de las lecciones del lecturer1.

- Tests de desempeño: Después de ejecutar los tests se han recopilado los datos del desempeño que ahora se van a desglosar:

  En una primera iteración se ejecutarón los tests para comprobar si todo funcionaba correctamenete, con los siguientes resultados:

Media	72,1778656
Error típico	5,61646002
Mediana	50
Moda	20
Desviación estándar	89,3352655
Varianza de la muestra	7980,78967
Curtosis	28,8121873
Coeficiente de asimetría	4,85495882
Rango	727
Mínimo	15
Máximo	742
Suma	18261
Cuenta	253
Nivel de confianza(95,0%)	11,061182

En resumen nuestro intervalo de confianza es:

	Bajo	Alto
Intervalo en ms	61,11668363	83,2390476
Intervalo en s	0,061116684	0,083239048

Luego después de algunas labores de mejora se volvieron a ejecutar los tests con los siguientes resultados:

Media	36,6600791
Error típico	2,91348141
Mediana	20
Moda	14
Desviación estándar	46,3417587
Varianza de la muestra	2147,5586
Curtosis	26,5918888
Coeficiente de asimetría	4,30837466
Rango	433
Mínimo	4
Máximo	437
Suma	9275
Cuenta	253
Nivel de confianza(95,0%)	5,73787544

Luego nuestro nuevo intervalo de confianza es:

	Bajo	Alto
Intervalo (ms)	30,9222036	42,3979545
Intervalo (s)	0,0309222	0,04239795

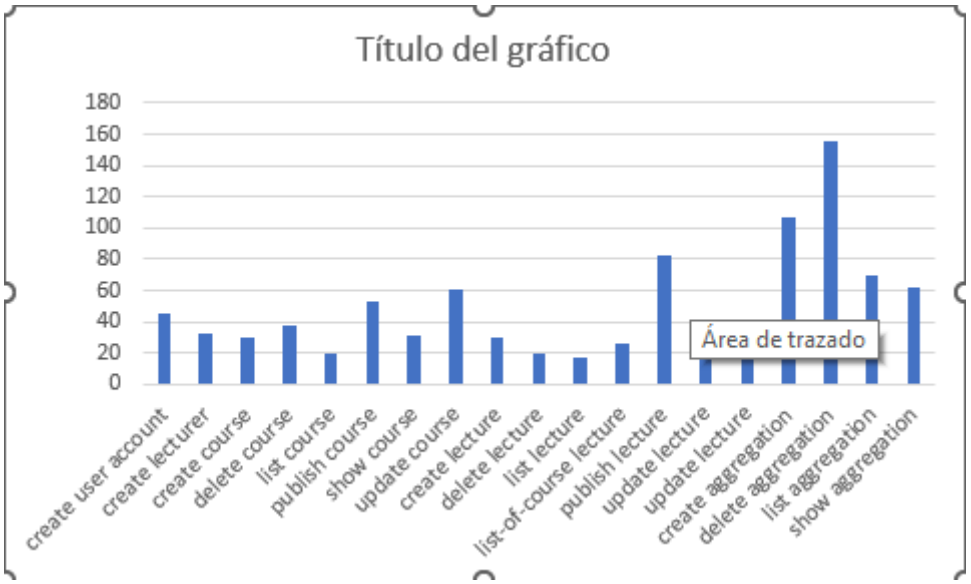
Luego comprobamos si hemos mejorado nuestros tiempos de respuesta:

	Antes	*Despues*
Media	72,1778656	36,6600791
Varianza (conocida)	7980,78967	2147,5586
Observaciones	253	253
Diferencia hipotética de las medias	0	

z	5,61354024	
P(Z<=z) una cola	9,91E-09	
Valor crítico de z (una cola)	1,64485363	
Valor crítico de z (dos colas)	1,98E-08	
Valor crítico de z (dos colas)	1,95996398	

Luego podemos decir que no existen evidencias suficientes para afirmar que la media de ambos tests sean los mismos y como el intervalo de confianza es observablemente mejor después de los cambios podemos decir que hemos mejorado el desempeño de la aplicación.

A continuación se muestra un desglose del tiempo medio en ms de los tiempos de cada una de las características:



Podemos denotar que nuestro mejor tiempo está en el listado de lecciones y que nuestro peor tiempo se corresponde con el borrado de las relaciones entre curso y lección.

5. Conclusión

Se puede decir que los tests juegan un papel importante a la hora de medir de forma sistemática el desempeño funcional y de respuesta de la aplicación ayudándonos a encontrar errores y cuellos de botella.

6. Bibliografía

No aplica.