

DP2 2023-2024

Testing report

Acme-SF-D04

Repositorio: <https://github.com/andpizcer/DP2-C1.032>

Miembro:

- Torralba Lanzas, Lorenzo, lortorlan1@alum.us.es

Tutor: Patricia Jiménez Aguirre

GRUPO C1.032 (G3 teoría)

27/05/2024

Historial de versiones

Fecha	Versión	Descripción de los cambios	Sprint
27/05/2024	V1	Creación del documento.	4

Índice

1.	Resumen Ejecutivo	4
2.	Introducción	4
3.	Pruebas funcionales	5
4.	Pruebas de rendimiento	8
5.	Conclusiones	12
6.	Bibliografía.....	12

1. Resumen Ejecutivo

En el presente documento se especificarán el documento de testing, se dividirá entre la explicación de las pruebas y los resultados obtenidos.

2. Introducción

A continuación, se detallarán en la primera parte, todos los casos de prueba que he realizado en el testing de las entidades implementadas. Se dividirá entre las dos entidades y a su vez entre las pruebas .safe y .hack. Tras esto, se hará un análisis del rendimiento de la aplicación en dos ordenadores diferentes, como así se indica en el anexo. Finalmente, se hará una comparación entre ambos.

3.Pruebas funcionales

Primeramente, voy a explicar las pruebas que he realizado para la entidad Training Module. Las pruebas correspondientes a los .safe han sido:

- List: Para el safe de list he listado los training modules de un developer.
- Show: En este caso he mostrado un training module publicado y otro sin publicar(para que se muestren todos los botones).
- Delete: En este caso simplemente he eliminado un training module.
- Update: Para update, primero he probado a actualizar sin datos, después con todos los datos válidos menos el code, después he ido probando con un code ya utilizado por otro módulo y con varios datos no válidos. Tras esto, el atributo detalles vacío, con 101 caracteres y ya con uno válido. El moduleDifficulty lo he probado poniéndolo a null solo, ya que al ser un select no se puede poner un valor no válido que no sea null. En el link he probado a ponerlo nulo, a ponerlo con un formato que no sea válido y a ponerle un link de mas de 255 caracteres. En el caso del total time, he probado a ponerle nulo, cero y un número negativo, además del 10001, todo ellos valores no válidos. Finalmente, el project solo lo he probado a null, ya que es un select también. Tras probar todos los casos malos, he añadido unos pocos de updates con datos válidos para testear bien los rangos.
- Publish: Para el publish, he seguido la misma metodología de update, para probar todos los valores posibles y que el coverage sea adecuado. Además, pruebo a publicar uno sin sesiones o con sesiones sin publicar, ya que ambos son casos de prueba negativos que se incluyen en el .safe.
- Create: Nuevamente he seguido la misma metodología probando todos los casos posibles antes de crear el training module.

A continuación, detallaré las pruebas que corresponden al .hack:

- List: Pruebo a listar sin haber iniciado sesión.
- Show: Pruebo a mostrar un módulo nulo sin haber iniciado sesión, mostrar uno que si exista sin haber iniciado sesión, mostrar uno nulo habiendo iniciado sesión y a mostrar uno que no me pertenece.
- Delete: Pruebo a eliminar sin iniciar sesión, a eliminar uno que no sea mío, tanto publicado como sin publicar, a eliminar uno publicado y a eliminar uno que no exista.
- Update: Pruebo a actualizar sin iniciar sesión, a actualizar uno que no me pertenece, uno que está publicado y uno que no existe.
- -Publish: Pruebo a publicar sin iniciar sesión, a actualizar uno que no me pertenece, uno que está publicado y uno que no existe.
- Create: Pruebo a crear sin haber iniciado sesión.

Para finalizar, pasamos con la entidad training session. En el caso del .safe:

- List: En este caso listo las sesiones de un módulo publicado y otro sin publicar (para que se muestre el botón créate).
- Show: Para show muestro una sesión publicada y otra sin publicar.
- Delete: Simplemente borro una training session.
- Update: Para probar el update vuelvo a seguir la metodología indicada en clase. Empiezo probando un code vacío, uno ocupado y varias formas para que no cumpla la expresión necesaria. Ahora para probar la fecha de comienzo y de fin, empiezo probando con la fecha de inicio a nulo, después pruebo una fecha que no cumpla con el espacio de una semana entre la creación del módulo, luego otra que sea después de una semana antes de la fecha de fin y por último una fecha por encima de la fecha límite. Para la fecha de final, pruebo una vacía, una fecha que sea menos de una semana después de la fecha de inicio y por último una fecha por encima del máximo. Para los atributos location e instructor son de tipo String y los pruebo con nulo y con 76 caracteres, ya que el máximo para ambos es 75. En el caso del email, pruebo con nulo, con un email no válido y con uno de más de 255 caracteres. Finalmente, el enlace lo pruebo con un enlace no válido y otro con más de 255 caracteres. Tras probar todos los datos erróneos, nuevamente se actualizan un par de sesiones más para probar con datos válidos como se indica en clase.
- Publish: En este caso, nuevamente se vuelve a realizar el proceso descrito arriba pero publicando entidades en vez de actualizándolas.
- Create: Al igual que los dos anteriores, se prueban todos los errores posibles y después se crean unas cuantas sesiones válidas.

Por último, las pruebas relacionadas con los .hack son:

- List: Pruebo a listar sin iniciar sesión las training sessions de un módulo que no existe. Después inicio sesión y pruebo las de un módulo que no existe y las de uno que no me pertenece.
- Show: Pruebo a mostrar sin iniciar sesión una que no existe e iniciando sesión una que no existe y otra que no me pertenece.
- Delete: Pruebo a eliminar sin iniciar sesión, a eliminar una que no sea mía, tanto publicada como sin publicar, a eliminar una publicada y a eliminar una que no exista.
- Update: Pruebo a actualizar sin iniciar sesión, a actualizar una que no me pertenece, una que está publicada y una que no existe.
- -Publish: Pruebo a publicar sin iniciar sesión, a actualizar una que no me pertenece, una que está publicada y una que no existe.
- Create: en este caso, sin iniciar sesión pruebo a crear una sesión para un módulo que no existe, uno publicado y otro sin publicar. Una vez he iniciado sesión pruebo a crear una para un módulo que no existe, uno que no me pertenece y otro que está publicado.

Tras completar todos los tests, no he encontrado ningún error grave, sin embargo, me ha servido para arreglar pequeños errores que hubiera sido complicado de hallar sin el debido testing.

4.Pruebas de rendimiento

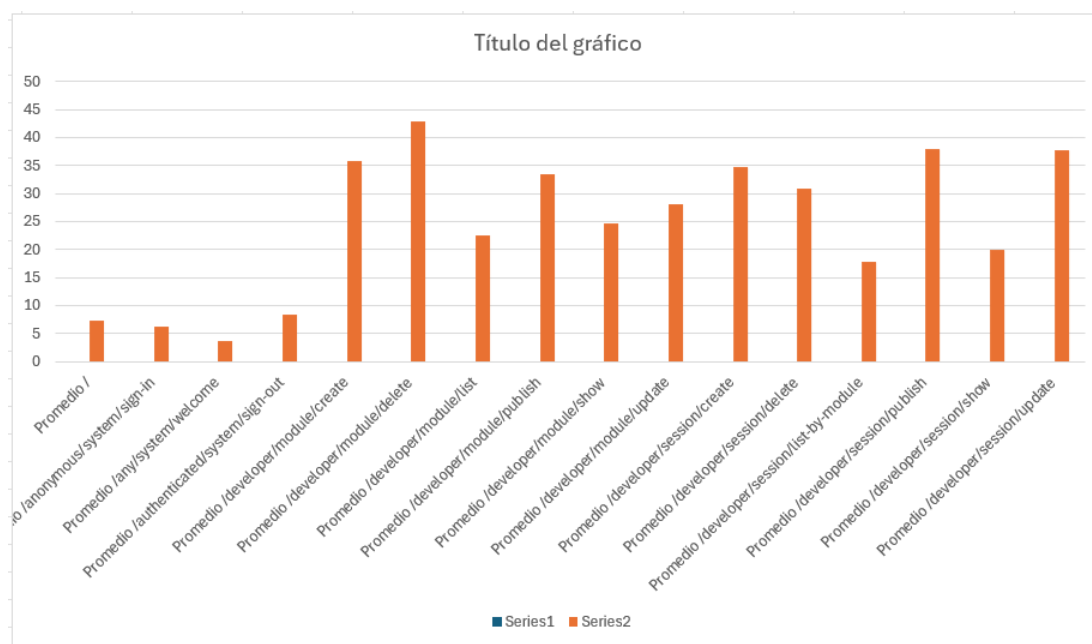
Los tests han sido probados en mi ordenador (Student3) y en los del Student2. En primer lugar, detallaré los resultados obtenidos en mi ordenador, cuyas especificaciones son las siguientes:

- **Procesador:** AMD Ryzen™ 5 7520U / 2.8Ghz.
- **Número de núcleos:** 4.
- **Almacenamiento principal:** 512GB SSD.
- **Memoria RAM principal:** LPDDR5 16GB.

A continuación, se expondrán los resultados obtenidos.

a) Promedio de peticiones y gráfica:

request-path	response-status	time
Promedio /		7,339785106
Promedio /anonymous/system/sign-in		6,30911875
Promedio /any/system/welcome		3,782870833
Promedio /authenticated/system/sign-out		8,4481
Promedio /developer/module/create		35,82041923
Promedio /developer/module/delete		42,94856667
Promedio /developer/module/list		22,45485429
Promedio /developer/module/publish		33,55874444
Promedio /developer/module/show		24,58500556
Promedio /developer/module/update		28,14704167
Promedio /developer/session/create		34,70434848
Promedio /developer/session/delete		30,9076
Promedio /developer/session/list-by-module		17,76375556
Promedio /developer/session/publish		37,87422963
Promedio /developer/session/show		19,99518889
Promedio /developer/session/update		37,74515556
Promedio general		21,83243404



b) Intervalo de confianza del 95%:

Columna1					
Media	21,83243404				
Error típico	0,885750544		interval(ms)	20,09143691	23,5734312
Mediana	21,3381		interval(s)	0,020091437	0,02357343
Moda	#N/D				
Desviación estándar	18,28168524				
Varianza de la muestra	334,2200153				
Curtosis	18,74924322				
Coefficiente de asimetría	2,806878227				
Rango	172,8447				
Mínimo	2,014				
Máximo	174,8587				
Suma	9300,6169				
Cuenta	426				
Nivel de confianza(95,0%)	1,740997132				

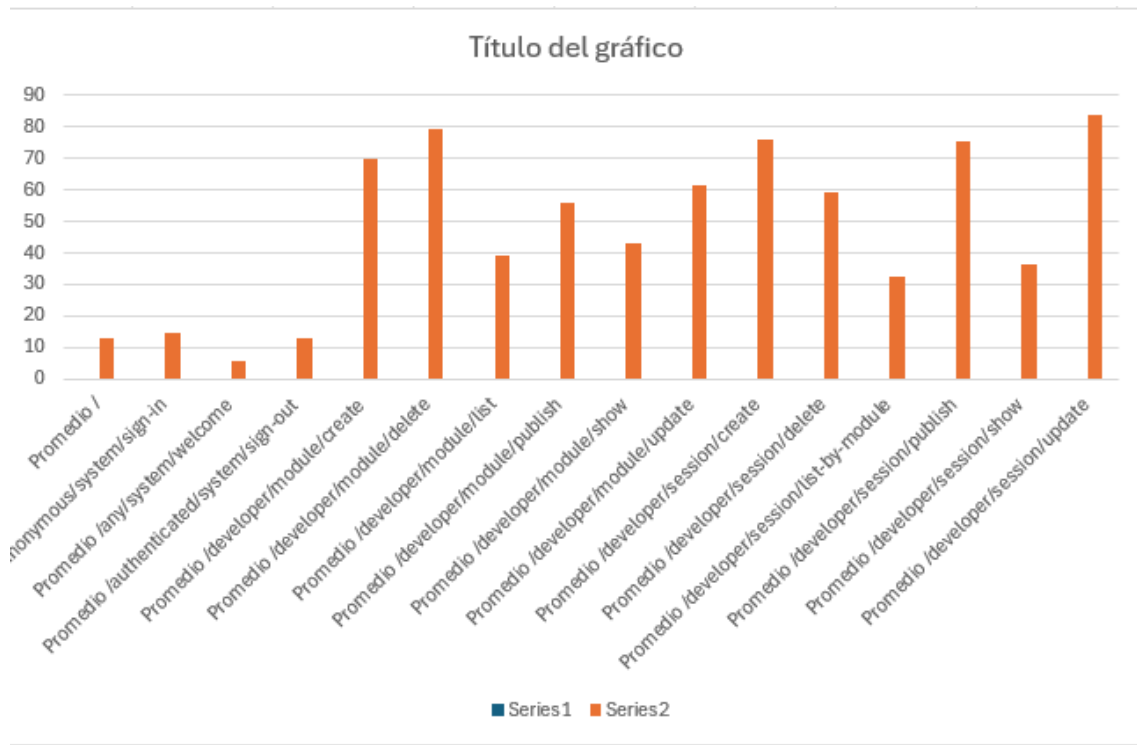
En segundo lugar, detallaré los resultados obtenidos en el ordenador del Student2, cuyas especificaciones son las siguientes:

- **Procesador:** Cpu Intel Core i5 4ª Gen 4210u / 1.7Ghz.
- **Número de núcleos:** Dual-core.
- **Almacenamiento principal:** 1Tb Hdd / 5400 Rpm.
- **Almacenamiento añadido:** 240Gb SSD.
- **Memoria RAM principal:** RAM 4Gb (1 X 4Gb), 1600Mhz.
- **Memoria RAM añadida:** RAM DDR3 4Gb, 1600Mhz.

A continuación, se expondrán los resultados obtenidos.

a) Promedio de peticiones y gráfica:

request-path	response-status	time
Promedio /		12,99969574
Promedio /anonymous/system/sign-in		14,51490625
Promedio /any/system/welcome		5,938804167
Promedio /authenticated/system/sign-out		12,6654
Promedio /developer/module/create		70,10740769
Promedio /developer/module/delete		79,18123333
Promedio /developer/module/list		39,11343714
Promedio /developer/module/publish		55,84634444
Promedio /developer/module/show		43,00436667
Promedio /developer/module/update		61,38322083
Promedio /developer/session/create		76,21124242
Promedio /developer/session/delete		59,22096
Promedio /developer/session/list-by-module		32,67267778
Promedio /developer/session/publish		75,34885556
Promedio /developer/session/show		36,22135556
Promedio /developer/session/update		84,10385556
Promedio general		42,62412418



b) Intervalo de confianza del 95%:

Columna1					
Media	42,62412418				
Error típico	1,980737957	interval(ms)	38,730862	46,5173864	
Mediana	35,4259	interval(s)	0,03873086	0,04651739	
Moda	#N/D				
Desviación estándar	40,88197079				
Varianza de la muestra	1671,335536				
Curtosis	12,1100483				
Coeficiente de asimetría	2,543268753				
Rango	356,254				
Mínimo	2,0705				
Máximo	358,3245				
Suma	18157,8769				
Cuenta	426				
Nivel de confianza(95,0%)	3,893262188				

Ahora observaremos el contraste de hipótesis del 95% para ver que terminal es más potente:

Prueba z para medias de dos muestras		
	<i>student 2</i>	<i>student 3</i>
Media	42,62412418	21,83243404
Varianza (conocida)	1671,335536	334,2200153
Observaciones	426	426
Diferencia hipotética de las medias	0	
z	9,582465092	
P(Z<=z) una cola	0	
Valor crítico de z (una cola)	1,644853627	
Valor crítico de z (dos colas)	0	
Valor crítico de z (dos colas)	1,959963985	

Para comenzar con el análisis del contraste de hipótesis, debemos verificar si el valor crítico de z (dos colas) se encuentra en el intervalo $[0, 0.05)$, dado que 0.05 es igual a 1 menos el nivel de confianza (95%). Como el valor crítico es 0, podemos comparar las medias de las solicitudes procesadas en ambos ordenadores. Como se puede observar, la media del ordenador de mi compañero(42,624) es la mitad de la de mi ordenador (21,8324), lo que indica que es significativamente más rápido. Esta diferencia se debe a las especificaciones de cada equipo, ya que el de mi compañero es ya bastante antiguo.

5. Conclusiones

En este informe se han detallado los procedimientos y resultados de las pruebas funcionales y de rendimiento realizadas en dos ordenadores con diferentes especificaciones. A través del análisis de los casos de prueba de las entidades Training Module y Training Session, se ha verificado la funcionalidad y la robustez de las aplicaciones, identificando y corrigiendo pequeños errores que podrían haber pasado desapercibidos sin un testing adecuado.

Además, el análisis de rendimiento comparativo entre dos ordenadores ha demostrado diferencias significativas en la eficiencia de procesamiento. El ordenador más reciente, con mejores especificaciones de hardware, ha resultado ser considerablemente más rápido que el ordenador más antiguo. Esta diferencia subraya la importancia de considerar las características técnicas al evaluar el desempeño de aplicaciones.

Finalmente, el contraste de hipótesis ha confirmado que la diferencia en el rendimiento no es fruto del azar, sino de los componentes de cada equipo. Estos resultados ponen de manifiesto la relevancia de realizar pruebas exhaustivas y de utilizar equipos adecuados para asegurar el óptimo funcionamiento de las aplicaciones en distintos entornos.

6. Bibliografía

Intencionalmente en blanco.