

Universidad de Sevilla

Escuela Técnica Superior de Ingeniería Informática

Testing Report



ACME SOFTWARE FACTORY

OUR FIRST PROJECT IN D&T

Grado en Ingeniería Informática – Ingeniería del Software


Diseño y Pruebas 2

Curso 2023 – 2024

Grupo de prácticas: C1-009


Autores por orden alfabético:

Carreño Mariño Ricardo

	Diseño y Pruebas II Acme-Software-Factory
	Testing Report


Índice de contenido

1. Resumen ejecutivo	3
2. Tabla de revisiones.....	4
3. Introducción	5
4. Contenido.....	6
5. Conclusiones	9
6. Bibliografía	11

	Diseño y Pruebas II Acme-Software-Factory
	Testing Report

1. Resumen ejecutivo

En este informe se presenta un análisis detallado de las pruebas y los resultados obtenidos para las funcionalidades correspondientes al módulo Student 1.

	Diseño y Pruebas II Acme-Software-Factory
	Testing Report

2. Tabla de revisiones

Fecha	Versión	Descripción
25/05/2024	1.0	Primera versión del documento.

	Diseño y Pruebas II Acme-Software-Factory
	Testing Report

3. Introducción

El documento se divide en dos partes principales: pruebas funcionales y pruebas de rendimiento. El objetivo de este informe es proporcionar un análisis detallado de los casos de prueba y del rendimiento de los mismo.

La primera parte del informe abarca las pruebas funcionales, las cuales se centran en evaluar las funcionalidades de tres entidades específicas: projects, userStories y madeOf (entidad intermedia entre Project y user Story). Se presentan los casos de prueba realizados, organizados por entidad, con descripciones claras y precisas para cada uno. Esto permite una evaluación concisa y efectiva de las funcionalidades asociadas a projects, userStories y madeOf.

En la segunda parte, el documento se centrará en proporcionar gráficos detallados que muestran los tiempos de respuesta del sistema durante las pruebas funcionales mencionadas anteriormente. Se comparará el desempeño del proyecto en dos entornos distintos, proporcionando un análisis más exhaustivo del rendimiento. Esta comparación permitirá identificar las diferencias en eficiencia y capacidad de manejo de carga entre los dos sistemas evaluados.

	Diseño y Pruebas II Acme-Software-Factory
	Testing Report

4. Contenido









Pruebas funcionales.


Requirement #6

Para este requisito, se han implementado las siguientes pruebas:

- List: En este test se ha comprobado que el listado de los projects se realiza de manera correcta.
- Show: Se ha comprobado que al entrar en un project se muestran los datos específicos de ese project correctamente.
- Update: Se comprueba que se realizan las actualizaciones de manera correcta, por otro lado, también se ha probado que al actualizar algún campo con valores incorrectos se muestre el error pertinente.
- Delete: Se ha comprobado que se puede eliminar un project sin ningún problema.
- Publish: Se comprueba que se puede publicar un project sin ningún impedimento, así como las restricciones al publicarlos sin cumplir todas las necesidades.
- Create: Se han comprobado todas las restricciones posibles en los campos de creación de un Project, a la vez que mandar el formulario vacío.
- Wrong-role: Este test comprueba que un usuario con otro rol que no sea manager no puede realizar las acciones de un manager.
- Wrong-id: Este test comprueba que no se pueden realizar acciones ya sea publicar, actualizar o borrar un proyecto con un ID no existente
- Right-role-Wrong-user: Se ha comprobado iniciando sesión como el manager2 no pueda acceder a los projects del manager1 para realizar acciones sobre estos.
- Published-project: Este test comprueba que no se pueden realizar acciones sobre un proyecto ya publicado

El cuanto al coverage, nos queda un total del 85,3% de la funcionalidad cubierta por los test.

acme.features.manager.project		85,3 %	1.137	196	1.333
> ManagerProjectDeleteService.java		68,1 %	213	100	313
> ManagerProjectPublishService.java		90,2 %	276	30	306
> ManagerProjectUpdateService.java		89,3 %	226	27	253
> ManagerProjectCreateService.java		89,9 %	204	23	227
> ManagerProjectShowService.java		91,4 %	127	12	139
> ManagerProjectListService.java		93,3 %	56	4	60
> ManagerProjectController.java		100,0 %	35	0	35

	Diseño y Pruebas II Acme-Software-Factory
	Testing Report










Como podemos observar la funcionalidad número 6 queda en prácticamente su totalidad, exceptuando sobre todo en el método de borrar el método unbind.

Requirement #7

Para este requisito, se han implementado las siguientes pruebas:

- List: En este test se ha comprobado que el listado de las user stories se realiza de manera correcta.
- Show: Se ha comprobado que al entrar en una user story se muestran los datos específicos de esa user story correctamente.
- Update: Se comprueba que se realizan las actualizaciones de manera correcta, por otro lado, también se ha probado que al actualizar algún campo con valores incorrectos se muestre el error pertinente.
- Delete: Se ha comprobado que se puede eliminar una user story sin ningún problema.
- Publish: Se comprueba que se puede publicar una user story sin ningún impedimento, así como las restricciones al publicarlas sin cumplir todas las necesidades.
- Create: Se han comprobado todas las restricciones posibles en los campos de creación de una user story, a la vez que mandar el formulario vacío.
- Wrong-role: Este test comprueba que un usuario con otro rol que no sea manager no puede realizar las acciones de un manager.
- Wrong-id: Este test comprueba que no se pueden realizar acciones ya sea publicar, actualizar o borrar una user story con un ID no existente
- Right-role-Wrong-user: Se ha comprobado iniciando sesión como el manager2 no pueda acceder a las user stories del manager1 para realizar acciones sobre estas.
- Published-project: Este test comprueba que no se pueden realizar acciones sobre una user story ya publicado

En cuanto al coverage, se ha realizado una cobertura del 86,9% de la funcionalidad gracias a los test.

acme.features.manager.userStory		86,9 %	1.041	157	1.198
> ManagerUserStoryDeleteService.java		63,2 %	117	68	185
> ManagerUserStoryCreateService.java		88,3 %	173	23	196
> ManagerUserStoryPublishService.java		90,0 %	208	23	231
> ManagerUserStoryUpdateService.java		89,4 %	195	23	218
> ManagerUserStoryShowService.java		91,9 %	137	12	149
> ManagerUserStoryListAllService.java		93,5 %	58	4	62
> ManagerUserStoryListService.java		96,5 %	111	4	115
> ManagerUserStoryController.java		100,0 %	42	0	42







Como podemos observar la funcionalidad número 7 queda prácticamente cubierta, exceptuando también el método de borrar el método unbind.

	Diseño y Pruebas II Acme-Software-Factory
	Testing Report

Para la entidad intermedia, se han implementado las siguientes pruebas:

- List: En este test se ha comprobado que el listado de las asignaciones se realiza de manera correcta.
- Show: Se ha comprobado que al entrar en una asignación se muestran los datos específicos de esa asignación correctamente.
- Create: Se han comprobado todas las restricciones posibles en los campos de creación de una asignación, a la vez que mandar el formulario vacío.
- Delete: Se ha comprobado que se puede eliminar una asignación sin ningún problema.
- Right-role-Wrong-user: Se ha comprobado iniciando sesión como el manager2 no pueda acceder a las asignaciones del manager1 para realizar acciones sobre estas.
- Wrong-id: Este test comprueba que no se pueden realizar acciones sobre una asignación con un ID no existente

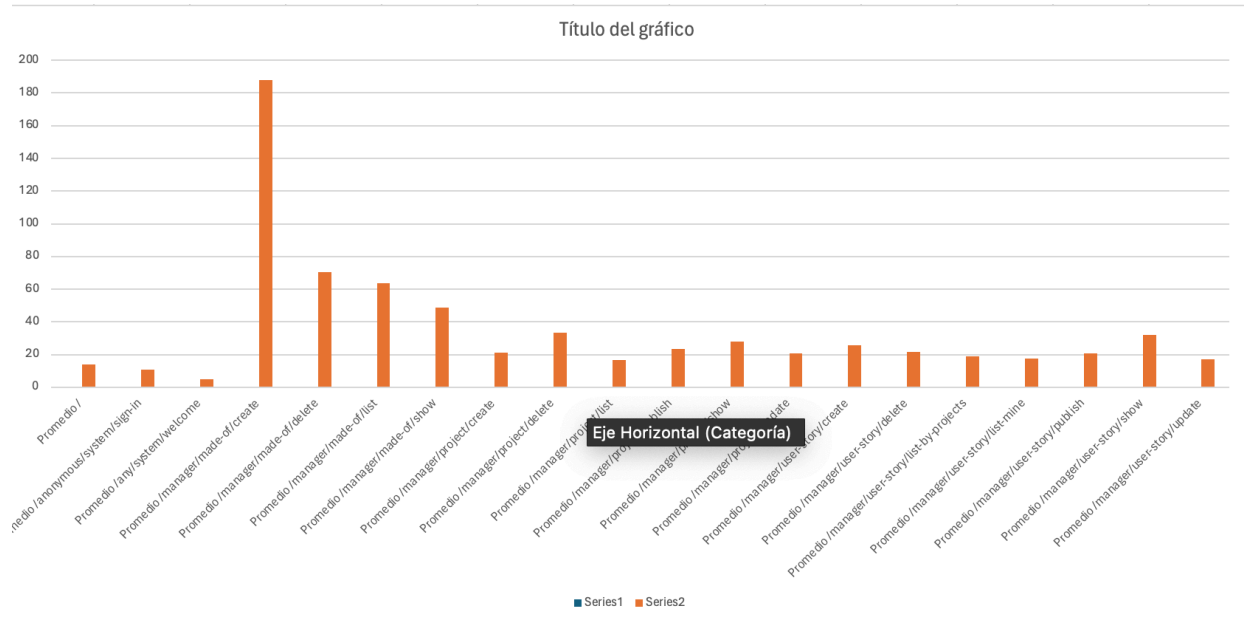
En cuanto al coverage, se ha realizado una cobertura del 81,8% de la funcionalidad gracias a los test.

acme.features.manager.madeOf		81,8 %	549	122	671
> ManagerMadeOfDeleteService.java		50,3 %	90	89	179
> ManagerMadeOfCreateService.java		90,8 %	227	23	250
> ManagerMadeOfShowService.java		95,6 %	131	6	137
> ManagerMadeOfListService.java		95,1 %	77	4	81
> ManagerMadeOfController.java		100,0 %	24	0	24

Como podemos observar la funcionalidad de la entidad intermedia queda prácticamente cubierta, exceptuando también el método de borrar el método unbind.

Pruebas de rendimiento.


Se ha realizado un estudio del rendimiento de las pruebas anteriores, se ha generado esta gráfica para observar claramente cuáles son las pruebas más ineficientes.



Como podemos observar, las peticiones más ineficientes es claramente el create de la entidad intermedia.


También se ha calculado el intervalo de confianza del 95 % para el tiempo que tarda en atender las solicitudes anteriores mi portátil. Nos quedaría:

Interval (ms)	15,3667583	21,2219974
Interval (s)	0,01536676	0,021222

	Diseño y Pruebas II Acme-Software-Factory
	Testing Report

5. Conclusiones

En conclusión, se ha cubierto casi en su totalidad las funcionalidades requeridas mediante las pruebas mencionadas. Los casos de prueba realizados han demostrado que tanto las entidades **project** como **userStory** y **madeOf** funcionan correctamente bajo las condiciones evaluadas. Cabe destacar que a lo mejor las métricas no son las mejores, pero se tiene que tener en cuenta que el realizador de estos tests los ha desarrollado mediante una máquina Virtual, lo cual nos saca el rendimiento completo del ordenador.

	Diseño y Pruebas II Acme-Software-Factory
	Testing Report

6. Bibliografía

<https://ev.us.es/>