

Diseño y Pruebas II



ANALYSIS REPORT D03

Grupo: C1.02.07

Repositorio: <https://github.com/DP2-C1-02-07/Acme-L3-D03>

Autor/es:

- Javier Nunes Ruiz - javnunrui@alum.us.es

Fecha: 21/04/2023

1. Índice

1. Índice	2
2. Tabla de versiones	3
3. Resumen del documento	3
4. Introducción	3
5. Contenidos	4
5.1 Operaciones de lecturers en courses	4
5.2 Operaciones de lecturers en lectures	6
5.3 Operaciones CourseLecture	8
6. Conclusiones	10
7. Bibliografía	10

2. Tabla de versiones

Versión	Fecha	Descripción
1.0.0	13/04/2023	Documento creado
1.0.1	14/04/2023	Finalización del documento
1.0.2	21/04/2023	Últimos retoques del documento

3. Resumen del documento

En este documento se aportará un listado de registros de análisis, cada uno de los cuales corresponden a un requisito de los propuestos en la entrega para el Student 1. Se llevarán a cabo todos los requisitos propuestos, pero únicamente se incluirán aquí aquellos requisitos que se consideren que requieren de un cierto análisis.

4. Introducción

Con este documento se pretende llevar a cabo un análisis exhaustivo de aquellos requisitos que consideremos que lo requieren, realizando una descripción detallada de los mismos, identificando las diferentes alternativas (si las hubiera) en que puede abordarse cada uno de ellos y las ventajas y desventajas que suponen escoger cada una de ellas, concluyendo con la alternativa finalmente seleccionada.

Para su realización se ha seguido la siguiente estructura: una portada con información sobre la persona que lo ha realizado e información del grupo al que pertenece, un resumen, una introducción, el contenido donde se detallan todos los aspectos del documento, una conclusión y finalmente la bibliografía consultada.

5. Contenidos

Los diferentes requisitos a analizar son:

5.1 Operaciones de lecturers en courses

Operations by lecturers on courses:

- List the courses that they have created.
- Show the details of their courses.
- Create, update, or delete their courses. Courses can be updated or deleted as long as they have not been published. For a course to be published, all of its lectures must have been published.

Requisito 14 del documento de requisitos: se trata de un requisito funcional cuyo objetivo es especificar las funciones que el sistema debe ser capaz de realizar para cumplir con los objetivos y necesidades del usuario.

En este caso, nos dicen que para que un curso sea publicado todas sus lecciones deben haber sido publicadas.

Tanto en este requisito como en aquellos donde se hable de publicar, tendremos en cuenta el atributo boolean `draftMode` (modo borrador) que lo incluiremos en las entidades `course` y `lecture`. Este atributo nos indica el estado en el que se encuentra un curso y una lección. Si toma el valor `true` significa que todavía se encuentra en modo borrador y puede ser actualizado y eliminado, por lo tanto aún no se ha publicado, mientras que si toma valor `false` ya se encuentra publicado y por ello no puede ser actualizado ni borrado.

Por lo que siguiendo con lo que estábamos analizando del requisito, analizamos las siguientes alternativas:

1. Que directamente no nos aparezca el botón de publicar si todas las lecciones del curso no han sido ya publicadas.

Ventajas: evita que aparezcan botones de más si aún no se podría accionar.

Desventajas: no notifica al usuario del por qué no saldría el botón de publicar y de lo que tiene que hacer para poder publicar un curso, podría pensar que se trata de un error de la aplicación, más difícil de implementar.

2. Que aparezca el botón de publicar y si todavía no se han publicado todas las lecciones, que al darle al botón nos deniegue la acción notificándonos el por qué con un mensaje.

Ventajas: nos deniega la acción a la vez que nos notifica el por qué, por lo que sabemos que hacer para poder publicar el curso.

Desventajas: podamos publicar o no un curso, nos sale el botón de publicar.

Por tanto, elegimos la segunda alternativa. Para ello, incluiremos en el método “validate” del servicio de publicar correspondiente la restricción donde se compruebe si todas las lecciones del curso que quiere publicarse ya se han publicado.

Por otro lado, para cumplir con una parte del requisito 6 de la entrega D02 que dice: “Purely theoretical courses must be rejected by the system” lo implementaremos de la misma forma, en el mismo método validate donde se compruebe que no todas las lecciones pertenecientes al curso que se quiere publicar sean teóricas. El sistema nos impedirá publicar el curso en el caso de que todas las lecciones sean teóricas, notificándonos con un mensaje.

A raíz de la implementación de las anteriores restricciones, surgió un nuevo análisis de una funcionalidad que no está en los requisitos: que no pueda publicarse un curso si no tiene lecciones asociadas, ya que es algo que realmente no tendría sentido. Por ello, de la misma forma que anteriormente incluimos en el mismo validate una restricción que compruebe que el curso que quiere publicarse tiene lecciones asociadas.

5.2 Operaciones de lecturers en lectures

Operations by lecturers on lectures:

- List the lectures in their courses.
- Show the details of their lectures.
- Create and publish a lecture.
- Update or delete a lecture as long as it is not published.

Requisito 15 del documento de requisitos: se trata de un requisito funcional cuyo objetivo es especificar las funciones que el sistema debe ser capaz de realizar para cumplir con los objetivos y necesidades del usuario.

Nos dicen que tenemos que listar las lecciones en los cursos a los que pertenecen, además de poder realizar las distintas operaciones con cada una de las lecciones. Por ello, surgen distintas alternativas para poder llevar a cabo la implementación:

1. Mostrar las lecciones dentro de cada curso.

Ventajas: vemos claramente qué lecciones pertenecen al curso al acceder a este.

Desventajas: no se tiene una lista de todas aquellas lecciones que pertenecen al profesor autenticado.

2. Mostrar todas las lecciones del profesor en una lista.

Ventajas: se tiene en una lista todas aquellas lecciones que pertenecen al profesor autenticado.

Desventajas: es más complicado y menos intuitivo mostrar en la lista junto a los atributos de la propia lección, información del curso al que pertenece.

3. Mostrar las lecciones tanto dentro de cada curso como todas las lecciones del profesor en una lista.

Ventajas: se ve qué lecciones pertenecen a cada curso, además de poder visualizar el profesor en una lista todas aquellas lecciones que tiene.

Desventajas: más tedioso.

Por tanto, elegimos la tercera opción ya que es la más completa y debido a que curso es un agregado, que significa que las lecciones pueden existir sin necesidad de estar asociadas a ningún curso y que una misma lección puede estar asociada a varios cursos. Por ello, se gestionará por un lado las operaciones de los cursos y por otro lado la de las lecciones y si fuera necesario se podrá combinar uno con otro ([más adelante se hablará](#)).

Dentro de los detalles de cada curso se mostrará un botón de lecciones que al pulsarlo nos llevará a un listado de las lecciones que pertenecen a ese curso, siendo posible pulsar en alguna de ellas para mostrar sus detalles. Si además, su atributo draftMode se encuentra en true nos aparecerá los botones para realizar las operaciones de actualización y borrado. Como se mencionó, por otro lado accederemos al listado de todas las lecciones que pertenecen al profesor autenticado, apareciendo un botón para crear una nueva (que por la lógica definida se creará sin asociarse a ningún curso, teniendo luego la posibilidad de hacerlo si se cree oportuno). Además, podremos pulsar en alguna lección para ver sus detalles y si su atributo draftMode se encuentra en true poder actualizarla o borrarla.

A continuación se presenta un enlace de validación disponible en el tablero de discusión de la asignatura:

https://ev.us.es/webapps/discussionboard/do/message?action=list_messages&course_id= 63009_1&nav=discussion_board&conf_id= 303964_1&forum_id= 206215_1&message_id= 363498_1

5.3 Operaciones CourseLecture

En este caso, no se trata de un requisito como tal, pero es algo necesario para así poder relacionar los cursos y lecciones. Surgen las siguientes alternativas para su implementación:

1. Que entrando en un curso del apartado cursos se puedan asociar lecciones al curso correspondiente.

Ventajas: forma más natural de llevarlo a cabo.

Desventajas: muchos botones que pueden saturar al usuario.

2. Que entrando en una lección del apartado lecciones pueda asociarse a alguno de los cursos.

Ventajas: menos botones que puedan saturar al usuario.

Desventajas: forma menos natural de llevarlo a cabo, tengo que cambiarme de menú al de cursos para ver las lecciones que ya están añadidas y las que voy añadiendo.

Por tanto, elegimos la primera alternativa ya que es la que mejor encaja en el sistema. Además, de que según el requisito 15 se deben listar las lecciones asociadas a un curso pues qué mejor manera que en el mismo sitio en la aplicación haya un botón para listar las lecciones ya asociadas al curso y otro para poder asociarlas. De la misma manera, se dispondrá de otro botón para eliminar una lección de un curso (se elimina únicamente la relación).

La forma más viable de poder añadir y eliminar lecciones de cursos es mediante un desplegable. El curso al que accedemos y por tanto, con el que queremos realizar acciones sobre sus lecciones se queda su atributo "code" a fijo y seleccionamos en el desplegable la lección según estemos para añadir o eliminar.

A continuación se presenta un enlace de validación disponible en el tablero de discusión de la asignatura, que ha sido útil para poder llevar a cabo esta parte de la funcionalidad:

https://ev.us.es/webapps/discussionboard/do/message?action=list_messages&course_id=63009_1&nav=discussion_board&conf_id=303964_1&forum_id=206215_1&message_id=364720_1

Por último, en todos aquellos requisitos (13, 14, 15) propuestos donde se tenga que listar cualquier entidad surge el siguiente análisis:

1. Mostrar todos los atributos de la entidad.

Ventajas: en la misma lista se puede ver toda la información de la entidad.

Desventajas: perdería un tanto el sentido implementar la funcionalidad de show, aparece un símbolo azul + para mostrar los demás atributos por lo que el usuario tiene que hacer una acción para verlos todos y además proporciona información redundante si se implementa el show.

2. Mostrar únicamente tres atributos de la entidad.

Ventajas: muestra un resumen de la entidad sin saturar al usuario con botones adicionales de más detalles, tiene sentido implementar la funcionalidad de show.

Desventajas: se tiene que acceder a alguna de las entidades del listado para ver el resto de sus atributos.

Por lo tanto, para todos los requisitos donde sea necesario elegimos la segunda alternativa ya que resulta la opción más sencilla visualmente para el usuario y es cómo tiene más sentido cada una de las distintas operaciones de las entidades.

6. Conclusiones

La realización de este documento ha permitido establecer un marco para la planificación, seguimiento y control del proyecto ayudando a garantizar que se cumplan los requisitos del proyecto de la mejor forma posible para así garantizar la mayor calidad y éxito del mismo.

Analizarlos hace que consideremos alternativas que con una simple lectura quizás no seamos capaces de identificar, pero que pueden incluso ser la mejor opción de implementación.

Requisitos de esta entrega como “Produce an analysis report” y “Produce a planning report” ya fueron analizados en la primera entrega por lo que las alternativas seleccionadas se continúan usando.

Con la realización de este reporte se ha podido comprobar que pueden surgir análisis de funcionalidades que no están explícitamente en los requisitos, pero que pueden ser más o menos importantes.

7. Bibliografía

- Documento “08 Annexes” de la asignatura Diseño y Pruebas 2, Universidad de Sevilla