

Universidad de las Ciencias Informáticas

Facultad 4



Módulo para la gestión de recursos educativos en RHODA

**Trabajo de Diploma para optar por el título de Ingeniero en Ciencias
Informáticas**

Autores:

Nadia Sánchez Batista

Pablo Reyes Díaz

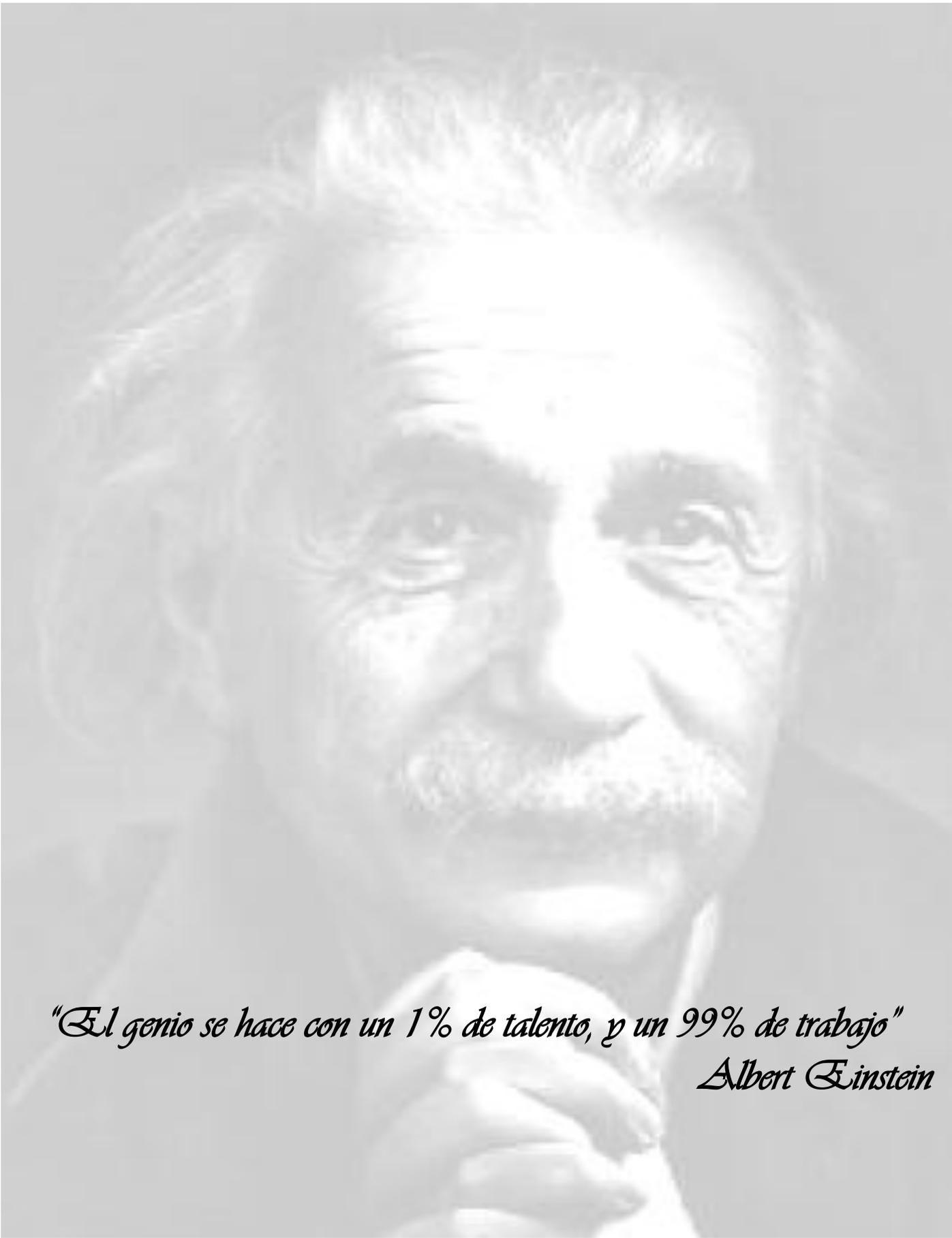
Tutores

Ing. Yailen San Juan Santana

Ing. Leonardo Rodríguez González

La Habana, junio de 2014

“Año 56 de la Revolución”



"El genio se hace con un 1% de talento, y un 99% de trabajo"
Albert Einstein

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas; para que haga el uso que estime pertinente.

Para que así conste, firmamos la presente a los __ días del mes de junio del año 2014.

AUTORES

Nadia Sánchez Batista

Pablo Reyes Díaz

TUTORES

Ing. Yailen San Juan Santana

Ing. Leonardo Rodríguez González

Dedicatoria

De Nadia

Este trabajo va dedicado a mis padres y hermano.

De Pablo

A mis padres y hermano.

Agradecimientos

Nadia

A mis padres y a mi hermano por su apoyo incondicional.

A Eduardo por creer en mí siempre, por ayudarme y apoyarme en los momentos más difíciles y por acompañarme durante estos 5 años.

A Pablo mi compañero de tesis por tener paciencia conmigo y por todas las noches en vela trabajando en la tesis.

A nuestros tutores Yailén San Juan Santana, Leonardo Rodríguez González y alguien que dice no ser tutor, pero siempre nos ayudó a Yandris Mata Cabrera.

A mis compañeras de apartamento por su apoyo y comprensión.

Pablo

A mis abuelos maternos y paternos, en especial a mi abuelita Irma que aunque ya no está físicamente siempre será mi guía.

A mis padres, a mi hermano y toda mi familia por el apoyo que siempre me dando.

A Nadia, mi compañera de tesis y de todos estos años en la universidad.

A mi tía Margara y su esposo Colomé por quererme y ayudarme en todo momento.

A mis tutores Yailén, Leonardo y Yandris.

A mi amiga Aymeé y a su familia por todo su apoyo incondicional durante estos cinco años.

A mis compañeros de apartamento en especial al Moro y Eduardo.

A mi amigo Aliesky por ayudarme en todos los momentos que necesité.

A mis amigos y hermanos de la infancia, por confiar en mí, Frank, Leandro y José.

A todas las persona con las he compartido durante estos 5 años.

Resumen

La introducción de los avances tecnológicos en la esfera educacional ha provocado un incremento en la producción de contenidos educativos en formato digital, haciendo necesaria la existencia de aplicaciones que permitan su almacenamiento. Diversos son los repositorios creados en Cuba, siendo RHODA, el repositorio de la Universidad de las Ciencias Informáticas, un ejemplo de ellos. A pesar de sus numerosas funcionalidades, RHODA solo es capaz de almacenar los recursos que son diseñados como objetos de aprendizaje, aspecto que impide centralizar los restantes contenidos educativos creados por la comunidad universitaria. La presente investigación tiene como objetivo desarrollar un módulo que permita la gestión de cualquier tipo de recurso educativo en RHODA. Para guiar el desarrollo se utilizó la metodología Rational Unified Process, así como la herramienta Visual Paradigm 8.0 y UML para el modelado. El proceso de implementación se llevó a cabo utilizando las herramientas Symfony 2.3.7, NetBeans 7.3.1, PostgreSQL 9.1 y Bootstrap 3.0. Además, se emplearon los lenguajes de programación: PHP5, CSS3, HTML5 y JavaScript 1.3. Se utilizaron de igual modo, la librería jQuery 1.10 y el servidor web Apache. El módulo desarrollado permite a los profesores de la universidad la gestión, visualización, descarga y almacenamiento de sus recursos. Contribuye además, a elevar la calidad de los mismos, como resultado de las valoraciones y sugerencias que pueden ser emitidas por otros usuarios. Por último, fomenta la reutilización, durabilidad y accesibilidad de los recursos respaldada por la utilización de estándares para su empaquetamiento y catalogación como: SCORM, IMS CP y LOM.

Palabras clave: estándar, gestión, objeto de aprendizaje, recurso educativo.

Índice

Introducción 1

Capítulo 1. Fundamentación teórica 5

 Introducción 5

 1.1 Recursos educativos 5

 1.2 Repositorios digitales 7

 1.2.1 Repositorios de recursos educativos 8

 1.2.2 Estudio de repositorios de recursos educativos similares 10

 1.3 Estándares y especificaciones de contenidos utilizados en las versiones anteriores de RHODA 11

 1.3.1 IMS Content Packaging 12

 1.3.2 IEEE/Learning Object Metadata 14

 1.3.3 Modelo Referenciado de Objetos de Contenido Compartible 15

 1.4 Metodología de desarrollo de software 16

 1.5 Tecnologías, lenguajes y herramientas 16

 1.5.1 Herramienta CASE 17

 1.5.2 Lenguaje de modelado 17

 1.5.3 Lenguaje de programación del lado del servidor 17

 1.5.4 Lenguajes de programación del lado del cliente 18

 1.5.5 Framework de desarrollo 19

 1.5.6 Entorno de desarrollo 20

 1.5.7 Sistema gestor de base de datos 21

 1.5.8 Servidor web 21

 1.6 Arquitectura de software 22

 Conclusiones parciales 22

Capítulo 2. Propuesta de solución 24

 Introducción 24

 2.1 Modelo del dominio 24

 2.2 Requisitos funcionales y no funcionales 25

 2.2.1 Requisitos funcionales 25

 2.2.2 Requisitos no funcionales 28

 2.3 Identificación de los actores y casos de uso del sistema 30

 2.3.1 Casos de uso del sistema 30

2.3.2 Diagrama de casos de uso del sistema	32
2.3.3 Descripciones de los CU	33
Conclusiones parciales	38
Capítulo 3. Análisis y diseño	39
Introducción	39
3.1 Modelo de análisis	39
3.1.1 Diagramas de clases del análisis	39
3.1.2 Diagramas de colaboración	42
3.2 Modelo del diseño	43
3.2.1 Patrones de diseño utilizados	43
3.2.2 Diagramas de clases del diseño	45
3.3 Modelo de datos	48
Conclusiones parciales	49
Capítulo 4. Implementación y pruebas	50
Introducción	50
4.1 Estándar de codificación	50
4.2 Modelo de implementación	50
4.2.1 Diagrama de componentes	51
4.2.2 Modelo de despliegue	52
4.3 Pruebas	53
4.3.1 Pruebas de caja negra	53
4.4 Resultados de las pruebas	57
Conclusiones parciales	58
Conclusiones generales	59
Recomendaciones	60
Referencias bibliográficas	61
Anexo I. Descripción de los casos de uso	65
Anexo II. Diagramas de clases del análisis	97
Anexo III. Diagramas de colaboración	104
Anexo IV. Diagramas de clases del diseño	110

Índice de figuras

Figura 1. Código de un archivo XML Manifiesto (16).....	13
Figura 2. Modelo de dominio.....	25
Figura 3. Modelo de extensión.....	32
Figura 4. Diagrama de CU del sistema.....	33
Figura 5. Estereotipo que representa la CI.....	40
Figura 6. Estereotipo que representa la CE.....	40
Figura 7. Estereotipo que representa la CC.....	40
Figura 8. DCA del CU: Importar recurso educativo.....	40
Figura 9. DCA del CU: Descargar recurso educativo.....	41
Figura 10. DCA del CU: Recomendar recurso educativo.....	41
Figura 11. DC del CU: Importar recurso educativo.....	42
Figura 12. DC del CU: Descargar recurso educativo.....	42
Figura 13. DC del CU: Recomendar recurso educativo.....	43
Figura 14. Ejemplo de utilización del patrón Alta Cohesión.....	44
Figura 15. Ejemplo de utilización del patrón Creador.....	45
Figura 16. DCD del CU: Importar recurso educativo.....	46
Figura 17. DCD del CU: Descargar recurso educativo.....	47
Figura 18. DCD del CU: Recomendar un recurso educativo a otro usuario.....	48
Figura 19. Modelo de datos.....	49
Figura 20. Diagrama de componentes.....	52
Figura 21. Modelo de despliegue.....	53
Figura 22. Resultados de las iteraciones.....	58

Índice de tablas

Tabla 1. Descripción de los actores del sistema	30
Tabla 2. Descripción del CU: Importar recurso educativo.....	34
Tabla 3. Descripción del CU: Descargar recurso educativo	36
Tabla 4. Descripción del CU: Recomendar un recurso educativo a otro usuario	36
Tabla 5. DCP del CU: Recomendar recurso educativo a otro usuario.....	55
Tabla 6. Descripción de las variables del CU: Recomendar recurso educativo a otro usuario.....	57
Tabla 7. Cantidad de no conformidades por iteración	57

Introducción

Las Tecnologías de la Información y las Comunicaciones (TIC), se introducen en la vida de los ciudadanos a una velocidad vertiginosa. Su desarrollo a partir de los avances científicos, ha posibilitado la creación de nuevas herramientas que ayudan y flexibilizan el trabajo del hombre.

En la actualidad, las TIC se convierten en un instrumento indispensable en las instituciones docentes, sirviendo de apoyo al proceso educativo, ya sea brindando acceso a la información digital, fomentando la comunicación entre estudiantes y profesores o apoyando la gestión del aprendizaje. De igual modo, posibilitan la creación de entornos virtuales, los cuales persiguen entre otros objetivos eliminar las restricciones que imponen el tiempo y el espacio entre profesores y estudiantes, dando paso al desarrollo del aprendizaje electrónico o e-learning.

El término e-learning describe el *“conjunto de tecnologías, aplicaciones y servicios orientados a facilitar la enseñanza y el aprendizaje a través de Internet/Intranet, que facilitan el acceso a la información y la comunicación con otros participantes”* (1), lo que permite que los estudiantes logren un aprendizaje menos dependiente de la figura del profesor, ya que interactúan en mayor medida con las tecnologías, favoreciendo la distribución y el intercambio de contenidos.

La aceptación del e-learning como forma de enseñanza y aprendizaje aumenta cada vez más, reportando como consecuencia un incremento de los recursos digitales que se crean con fines educativos. Estos recursos suelen estar constituidos por materiales, cursos, libros de texto, videos, exámenes, además pueden ser utilizados, compartidos, combinados y adaptados según las necesidades educativas.

Debido a la diversidad de los contenidos que se crean, resulta imprescindible que los mismos cumplan con una estructura o formato común, de manera tal que puedan ser manejados por las diferentes herramientas e-learning. Dicha estructura es proporcionada por los estándares de empaquetamiento IMS CP y SCORM, los cuales sirven de facilitadores de la durabilidad y la reutilización de los recursos. Ambos estándares recomiendan el uso de la especificación LOM para describir la información adicional que se añade a los contenidos y que tiene como objetivo describir sus características semánticas, así como facilitar los procesos de búsqueda y localización, tanto en repositorios como sistemas de gestión de aprendizaje y herramientas de autor.

Los repositorios en particular, constituyen las aplicaciones informáticas dedicadas al almacenamiento y manejo de recursos educativos, garantizando la accesibilidad, localización y reutilización de los mismos. La doctora Roxana Cañizares define a los repositorios como: *“un sistema especializado en el almacenamiento de recursos educativos, accesibles y operables por usuarios a través de Internet/Intranet,*

con funciones encaminadas a la clasificación, localización, reutilización, recuperación y mantenimiento de los recursos, que permita compartir estos con otras herramientas de un entorno b-learning". (2)

Los repositorios de recursos educativos facilitan el acceso a la información, proporcionando a las instituciones educacionales la oportunidad de compartir materiales en función del mejoramiento del aprendizaje. En Cuba, algunas instituciones han incorporado repositorios como herramientas de apoyo a sus procesos docentes, tal es el caso de la Red de Salud Cubana Infomed, La Universidad de la Habana, la Universidad Agraria de La Habana, la Universidad de Matanzas, el Instituto Superior Politécnico José Antonio Echeverría (CUJAE), la Universidad de Oriente y la Universidad de las Ciencias Informáticas (UCI), quien hace uso de un repositorio llamado RHODA. Este repositorio se encarga de almacenar, visualizar y gestionar los Objetos de Aprendizaje (OA) que se crean por parte de la comunidad universitaria, para su posterior utilización en el proceso de formación.

RHODA, actualmente, no permite el almacenamiento y gestión de cualquier tipo de recurso educativo, debido a que en su versión inicial fue concebido únicamente para el manejo de OA. Esta limitante trae consigo que no se disponga de un único sistema en el cual se centralicen todos los recursos, dígame cursos, multimedia e imágenes creados por los profesores de la Universidad, impidiendo así su localización y reutilización, provocando además, que otras herramientas se dediquen a estas acciones.

Para sentar las bases de la solución a este problema y lograr a su vez la integración de las diferentes aplicaciones con fines educativos que se desarrollan en la Universidad (CRODA, Zera), y delimitar además sus respectivas funcionalidades, se definió un nuevo marco de trabajo llamado Xalix. Sin embargo, la utilización de Xalix implica una nueva problemática y es la necesidad de implementar nuevamente las funcionalidades de RHODA, mediante el empleo del *framework* Symfony en su versión 2. Esta nueva versión representa cambios en el desarrollo de las aplicaciones, los que radican fundamentalmente en la concepción de las mismas como un conjunto de componentes independientes y fácilmente integrables. Por tal motivo, se hizo necesario realizar una nueva concepción del módulo de gestión de OA que presenta RHODA, con el objetivo de eliminar los problemas de incompatibilidad existentes con el nuevo marco de trabajo, así como las limitantes relacionadas con el manejo de recursos educativos.

A partir de la problemática antes descrita, se identifica como **problema de investigación**: ¿Cómo facilitar la gestión de los recursos educativos en RHODA?

Se plantea como **objeto de estudio**: la gestión de recursos educativos en repositorios digitales.

El **objetivo general** de la investigación es: desarrollar un módulo que permita la gestión de los recursos educativos en RHODA. El objetivo general se desglosa en los siguientes **objetivos específicos**:

- Elaborar el marco teórico-conceptual relacionado a la gestión de recursos educativos en los repositorios digitales.
- Diseñar las funcionalidades del módulo de gestión de recursos educativos en RHODA.
- Implementar el módulo de gestión de recursos educativos en RHODA.
- Realizar pruebas al módulo desarrollado.

El **campo de acción** es: la gestión de los recursos educativos en RHODA.

Como **idea a defender** se tiene que: con el desarrollo de un módulo para la gestión de los recursos educativos en RHODA, se contribuirá a la centralización y reutilización de los recursos creados por los profesores de la Universidad de las Ciencias Informáticas. Para dar cumplimiento a los objetivos antes mencionados, se proponen las siguientes **tareas de investigación**:

- Revisión bibliográfica relacionada al proceso de gestión de recursos educativos en los repositorios digitales.
- Selección y conceptualización de los elementos teóricos referentes al estado del arte de los recursos y repositorios digitales.
- Definición de los requerimientos funcionales y no funcionales del módulo para la gestión de recursos educativos en RHODA.
- Selección de las herramientas y tecnologías a utilizar.
- Diseño de las funcionalidades del módulo.
- Selección de los métodos y técnicas de pruebas a utilizar para garantizar el correcto funcionamiento del módulo implementado.
- Ejecución de las pruebas definidas con anterioridad.

Métodos de la investigación científica:

Los métodos de la investigación científica que se utilizan en la presente investigación son los siguientes:

Histórico-Lógico: se utiliza con el fin de estudiar los antecedentes y tendencias de los repositorios de recursos educativos existentes a nivel nacional e internacional.

Analítico-Sintético: permite el estudio y análisis de la bibliografía referente al tema, así como la extracción de los elementos relacionados con la gestión de recursos educativos en repositorios digitales, lo que posibilita ofrecer una propuesta adecuada que solucione el problema planteado.

La modelación: se utiliza para representar la información generada durante los flujos de trabajo de análisis, diseño, implementación y prueba.

El presente trabajo se estructura de la siguiente forma:

Capítulo 1: Fundamentación teórica. En este capítulo se abordan los fundamentos teóricos de la investigación, destacando los aspectos relacionados con la gestión de los recursos educativos en repositorios digitales. Se describen las herramientas, tecnologías y metodología a utilizar en el desarrollo del módulo.

Capítulo 2: Propuesta de solución. Se describe la propuesta de solución desde el modelo de dominio que representa los conceptos importantes asociados al negocio. Se muestran además, los requerimientos funcionales y no funcionales. Se identifican los actores del sistema, se elabora el modelo de casos de uso y se realizan las descripciones correspondientes a cada uno de ellos.

Capítulo 3: Análisis y diseño. Se describe la solución que se propone a partir de los diagramas de clases del análisis y de colaboración, asociados a las funcionalidades del sistema. Se abordan los aspectos relacionados con el diseño, incluyendo los patrones utilizados, evidenciando cómo se aplican cada uno de ellos en el mismo. Por último, se incluyen los diagramas de clases del diseño y el modelo de datos.

Capítulo 4: Implementación y pruebas. Se describen todos los elementos relacionados a la implementación, reflejando las prácticas de programación y los estándares de codificación utilizados. Además, se describe la estrategia trazada para desarrollar las pruebas al software, así como los resultados obtenidos luego de aplicada dicha estrategia.

Capítulo 1. Fundamentación teórica

Introducción

La gestión de contenidos en los entornos e-learning está generando cambios en diversos sentidos. Uno de estos cambios lo constituye la creación de los contenidos como recursos educativos y su almacenamiento en contenedores, conocidos como repositorios. Los repositorios de recursos educativos, permiten además la gestión, búsqueda y reutilización de los contenidos, haciendo uso de estándares reconocidos a nivel internacional.

1.1 Recursos educativos

Con el desarrollo de las TIC, el concepto de recurso educativo se ha visto enriquecido, vinculado al mismo existen diversas definiciones, incluso algunas bibliografías tratan este término como recursos digitales o recursos educativos abiertos.

El término Recursos Educativos Abiertos (REA), fue expuesto por la Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura (UNESCO) en el año 2002, con el objetivo de ofrecer de forma abierta recursos educativos provistos por medio de las TIC, para su consulta, uso y adaptación con fines no comerciales, son catalogados como: *“materiales de enseñanza, aprendizaje o investigación que se encuentren en el dominio público o que han sido publicados con una licencia de propiedad intelectual que permite su utilización, adaptación y distribución gratuita”*. (3)

Otras fuentes consultadas definen los REA como: *“recursos destinados para la enseñanza, el aprendizaje y la investigación que residen en el dominio público o que han sido liberados bajo un esquema de licenciamiento que protege la propiedad intelectual y permite su uso de forma pública y gratuita o permite la generación de obras derivadas por otros. Los Recursos Educativos Abiertos se identifican como cursos completos, materiales de cursos, módulos, libros, videos, exámenes, software y cualquier otra herramienta, materiales o técnicas empleadas para dar soporte al acceso de conocimiento”*. (4)

Otros autores consideran los REA como: *“materiales en formatos digitales que se ofrecen de manera gratuita y abierta para educadores, estudiantes y autodidactas para su uso y re-uso en la enseñanza, el aprendizaje y la investigación”*. (5)

“Los recursos educativos digitales son materiales compuestos por medios digitales y producidos con el fin de facilitar el desarrollo de las actividades de aprendizaje”. (6)

Las definiciones anteriores coinciden en que los recursos educativos son materiales digitales que facilitan la enseñanza y el aprendizaje de manera que puedan ser accesibles y gratuitos, teniendo en cuenta esta observación, en su tesis doctoral Cañizares propone: *“los recursos educativos están compuestos por contenidos educativos que pueden ser cursos completos, materiales para cursos, materiales multimedia, objetos de aprendizaje y cualquier otro recurso con fin educativo con una alta pertinencia y certificación de calidad. Son manejados por herramientas, ejemplo LMS¹ y repositorios, para su accesibilidad y disponibilidad, y a su vez, pueden ser distribuidos de forma gratuita y libre de restricciones o bajo una licencia”* (2). Después del análisis de los conceptos presentados, se adopta el enunciado anterior en la presente investigación teniendo en cuenta el nivel de generalización que se utiliza para identificar un recurso educativo.

Objetos de Aprendizaje

En el ámbito de la enseñanza, la corriente más destacada en cuanto al diseño de recursos educativos en formato digital es la de objetos de aprendizaje (7). Los OA son recursos educativos especiales, estos se caracterizan por cumplir determinadas pautas o requisitos.

Formalmente, no existe una definición única de OA, por el contrario, las definiciones son muy amplias, a continuación se analizan algunas de ellas:

En la bibliografía consultada, López Guzmán en su tesis doctoral en el año 2005, precisa como OA: *“cualquier recurso con una intención formativa, compuesto de uno o varios elementos digitales, descrito con metadatos, que pueda ser utilizado y reutilizado dentro de un entorno e-learning”.* (8)

Sin embargo, otros autores lo definen como: *“cualquier recurso digital que puede ser reutilizado para apoyar el aprendizaje”.* (7)

Otros afirman que un OA es un *“ recurso educativo digital relativamente pequeño y auto-contenido, con una marcada intención formativa, compuesto por uno o varios objetos de información, con un objetivo, descrito con metadatos y con un comportamiento secuenciado que asegure el correcto enlace entre los elementos de su estructura didáctica; concebido para diferentes poblaciones según sus contextos*

¹ Learning Management System: Sistema de Gestión de Aprendizaje, aplicación instalada en un servidor, que administra, distribuye y controla las actividades de formación de una institución u organización.

socioculturales para lograr su reutilización e interoperabilidad en varios entornos e-learning, que contribuya a la transmisión de conocimiento y la formación de valores tanto en los profesores, diseñadores, como en los usuarios finales” (9). Partiendo del análisis de los conceptos anteriores, se decide hacer uso de este último, para la presente investigación, pues se considera que contiene todo los elementos fundamentales que identifican un OA.

Los OA tienen una serie de características específicas que los diferencian de los materiales de aprendizaje tradicionales. Dichas características se refieren a aspectos técnicos, las cuales indican que los mismos deben ser (8):

- **Reutilizables:** el recurso debe ser modular para servir como base o componente de otro recurso. También debe tener una estructura y los componentes necesarios para ser incluido en diversas aplicaciones.
- **Accesibles:** pueden ser indexados para una localización y recuperación más eficiente, utilizando esquemas de estándares de metadatos.
- **Interoperables:** pueden ser manejados por diferentes plataformas de hardware y software.
- **Portables:** pueden intercambiarse y almacenarse en diferentes plataformas de manera transparente, sin cambio alguno en estructura o contenido.
- **Durables:** deben permanecer intactos a las actualizaciones de software y hardware.

Los recursos educativos de manera general, representan un factor fundamental en el aprendizaje electrónico, por lo que resulta necesario prever que los mismos puedan ser consultados y accedidos en el momento necesario. Para garantizar estos aspectos, así como su reutilización, los recursos se almacenan y gestionan en contenedores llamados repositorios. Estos sistemas permiten además, visualizar, exportar, importar y reemplazar recursos, así como realizar comentarios y sugerencias a los mismos.

1.2 Repositorios digitales

Según el diccionario de la Real Academia Española (REA) el término repositorio significa: “*Almacén o lugar donde se guardan ciertas cosas*” (10). Con el surgimiento de la era digital algunos autores sostienen la definición de repositorios digitales como: “*un sitio centralizado donde se almacena y mantiene información digital*”. (8)

Los repositorios digitales son básicamente sistemas en los que se guardan objetos digitales como vídeos, animaciones, imágenes, documentos y libros digitales. Estas aplicaciones organizan los objetos por categorías, áreas del conocimiento o temas. Sus objetivos esenciales son la búsqueda, acceso y visualización de materiales digitales, garantizando así la preservación y difusión de los mismos. En el mundo digital, los repositorios se clasifican en dos grandes grupos: *institucionales* y *temáticos*. (8)

“Los repositorios temáticos recogen documentos científicos y/o académicos de una o varias disciplinas científicas específicas y son los investigadores de diversas instituciones quienes contribuyen al autoarchivado de sus trabajos”. (11)

“Los repositorios institucionales a diferencia de los temáticos, reúnen la producción científica y/o académica de los miembros de una o varias instituciones, almacenando, preservando, divulgando y dando acceso abierto a los recursos depositados en ellos”. (11)

Teniendo en cuenta las definiciones anteriores, el curso de la presente investigación, se remonta a los repositorios institucionales, que como su concepto lo indica, permiten el almacenamiento de los recursos creados en una institución.

1.2.1 Repositorios de recursos educativos

Los repositorios de recursos educativos constituyen una distinción especial de los repositorios digitales y se distinguen por las características de los recursos que almacenan. (2)

Algunos autores definen a los repositorios de recursos educativos como: *“un tipo de repositorio que permite el almacenamiento, la gestión y el uso (reutilización) de recursos educativos”.* (7)

Otros autores utilizan el término de Repositorio Institucional de Recursos Educativos Digitales Abiertos y lo definen por ser: *“un sistema de información digital, definido, gestado, implementado y operado por una Institución de Educación Superior, cuya función principal es recopilar, almacenar, ordenar, localizar, preservar y redistribuir los recursos educativos digitales creados y producidos por la institución y los miembros de su comunidad, con la finalidad de disponerlos en un entorno web, de acceso público”.* (12)

En la presente investigación se toma como referencia el concepto que expone Cañizares donde especifica: *“un sistema especializado en el almacenamiento de recursos educativos, accesibles y operables por usuarios a través de Internet/Intranet, con funciones encaminadas a la clasificación,*

localización, reutilización, recuperación y mantenimiento de los recursos, que permita compartir estos con otras herramientas de un entorno e-learning". (2)

Los Repositorios Institucionales de Recursos Educativos Digitales Abiertos deben brindar los siguientes servicios (12):

- **Almacenamiento:** capacidad del repositorio que permite la gestión de alojamiento y preservación de los archivos de los recursos educativos digitales.
- **Búsqueda y recuperación:** función que permite la consulta, exploración y recuperación de información sobre los recursos que contiene el repositorio.
- **Navegación:** función que permite explorar los diferentes servicios, secciones y colecciones que se encuentran en el repositorio.
- **Visualización:** función que se establece para presentar la respuesta que se obtiene de la búsqueda.
- **Gestión de colecciones:** servicio que facilita el control, manejo y organización de la oferta de los recursos educativos digitales.
- **Gestión de usuarios:** servicio que permite la identificación del acceso al repositorio de diferentes tipos de usuarios según los privilegios otorgados.

Otros autores consultados, refieren que dentro de las funcionalidades básicas de los repositorios de recursos educativos deben encontrarse las siguientes (2):

- Importar un recurso educativo creado previamente por una herramienta de autor.
- Publicar un recurso educativo después de pasar por un proceso de revisión.
- Descargar los recursos educativos.

En la actualidad, la mayor parte de los repositorios existentes contienen las funcionalidades anteriores, algunos incluso, comprenden la creación de OA. En el siguiente epígrafe, se muestra un estudio realizado de diferentes repositorios con el objetivo de identificar las funcionalidades comunes referentes a la gestión de los recursos que almacenan.

1.2.2 Estudio de repositorios de recursos educativos similares

Alrededor del mundo, las instituciones educacionales de diversos países incorporan repositorios como herramientas de apoyo al proceso de enseñanza y aprendizaje, incrementando la interacción de los estudiantes con los medios electrónicos y las nuevas tecnologías. Numerosos son los repositorios reconocidos a nivel internacional por la calidad de los servicios que brindan y los recursos que almacenan.

❖ **Agrega**

Es un proyecto desarrollado entre el Ministerio de Educación, Cultura y Deporte, Red.es² y las Comunidades Autónomas en España. El objetivo de Agrega es facilitar a la comunidad educativa una herramienta útil que integre las TIC en el aula y fuera de ella, permitiendo además acceder a cualquier contenido, desde cualquier sitio y sea cual sea el criterio de búsqueda empleado. La plataforma es capaz de gestionar todo tipo de objetos utilizables en entornos web, de diferentes formatos digitales, ya sean objetos simples (texto, imágenes, archivos de audio, videos y otros) o cualquier combinación de los anteriores.

Agrega utiliza la especificación LOM y el estándar SCORM para la catalogación y el empaquetamiento de los contenidos respectivamente. Los contenidos a su vez, están curricularmente organizados de forma que puedan ser utilizados en la enseñanza no universitaria. Como parte de las funciones que brinda el repositorio, el usuario puede realizar valoraciones y comentarios, pre visualizar el contenido de cualquier recurso almacenado sin tener que estar autenticado en el sistema. Los recursos pueden ser descargados y exportados a diferentes formatos, además pueden ser enviados por correo electrónico a otros usuarios.

❖ **The Gateway**

Es un repositorio de recursos educativos de acceso público en los Estados Unidos, ha estado a disposición de los profesores de forma continua desde el año 1996 por lo que se considera el repositorio más antiguo (13). El mismo contiene diferentes tipos de recursos educativos para los docentes y estudiantes, desde actividades y planes de lecciones para proyectos en línea, hasta preguntas de evaluación. Sus recursos no solo pueden ser visualizados en la web, sino que también pueden ser descargados hacia cualquier dispositivo de almacenamiento con el fin de ser reutilizados.

² Red.es es una entidad pública empresarial adscrita al Ministerio de Industria, Energía y Turismo (MINETUR), que desarrolla un intenso conjunto de programas para que la sociedad española se beneficie al máximo de las posibilidades que ofrecen la TIC.

❖ **Repositorio de recursos educativos de la Universidad Virtual de Salud en Cuba**

El repositorio de recursos educativos de la Universidad Virtual de Salud (UVS) en Cuba, es un proyecto desarrollado por la red telemática Infomed. Este sistema tiene como objetivo garantizar la accesibilidad y la reutilización de recursos educativos producidos en el sistema nacional de salud, mediante el depósito, descripción y distribución de los mismos. La gestión de los recursos se realiza de forma centralizada, existiendo para esto un moderador que aprueba los recursos propuestos de forma descentralizada por los autores o especialistas encargados de la confección de los recursos. Esencialmente, el repositorio garantiza los servicios básicos de búsquedas simples y avanzadas, además de poder filtrar las mismas por los metadatos de los contenidos tales como: autor, título, tipo de recurso y año de creación. Igualmente, permite descargar e imprimir los recursos, así como recomendar los mismos a otros usuarios. Los usuarios una vez autenticados en el repositorio pueden calificar y comentar los recursos.

❖ **RHODA**

Es un Repositorio de Objetos de Aprendizaje (ROA), desarrollado por especialistas de la UCI. Tiene como objetivo fundamental almacenar y gestionar los OA que se crean por parte de la comunidad universitaria. Es una aplicación web modular y multiplataforma, que provee un lugar común accesible a través de un navegador, donde los usuarios pueden almacenar, recuperar y consultar OA, destinados a fortalecer el proceso de enseñanza-aprendizaje. El módulo dedicado a la gestión de los OA incorpora el estándar SCORM en sus versiones 1.2 y 1.3 (2004), además de la especificación LOM para la descripción de los mismos. Dicho módulo contiene funcionalidades que permiten a los usuarios visualizar, importar, calificar y descargar los OA. Ofrece además, los objetos más visualizados, descargados y solicitados.

El estudio de los repositorios anteriores, centrado en las funcionalidades que proporcionan referente a la gestión de los recursos que almacenan, permitió identificar las funcionalidades básicas que debe presentar el módulo a desarrollar, estas son: visualizar, listar, recomendar, comentar, importar, exportar y descargar los recursos educativos.

1.3 Estándares y especificaciones de contenidos utilizados en las versiones anteriores de RHODA

La Real Academia Española en su edición 2003, define un estándar como: “*un patrón, una tipificación o una norma de cómo realizar algo*” (10). Existen dos tipos de estándares: los estándares de jure y los estándares de facto. Los estándares de jure son aquellos que provienen de una organización acreditada

que certifica una especificación y que deben cumplir con tres niveles de trabajo: nivel de especificación, de validación y de estandarización, mientras los estándares de facto, se manifiestan cuando la especificación se adopta por un grupo mayoritario de individuos y que son usados por voluntad propia. (14)

En el ámbito de la educación en línea, los estándares adquieren cada vez mayor importancia, dada la relevancia que tienen para lograr una homogenización en la estructura de los recursos. El empleo de los mismos proporciona numerosas ventajas, las que radican fundamentalmente en la independencia de las plataformas, debido a que pueden crearse como paquetes auto-contenidos y ser intercambiados entre diferentes sistemas. (15)

Los estándares y especificaciones, mencionados en el epígrafe anterior, son utilizados para el empaquetamiento y descripción de los recursos educativos. A continuación se realiza un estudio de los estándares SCORM y LOM por la relevancia que presentan en la propuesta de solución.

1.3.1 IMS Content Packaging

IMS Content Packaging, es un estándar creado por la organización IMS Global Consortium Inc³, que tiene como objetivo, permitir la distribución de contenidos reutilizables e intercambiables. Ofrece una forma de empaquetar (en un archivo comprimido tipo Zip) los contenidos educativos tales como: cursos individuales, conjuntos de cursos o cualquier tipo de recurso necesario en el proceso educativo (por ejemplo, evaluaciones o exámenes). (16)

El archivo de intercambio de paquetes (PIF por sus siglas en inglés), es el archivo resultante de empaquetar todos los componentes del paquete de contenido asociados a los recursos, bajo la forma de un archivo comprimido. El PIF alberga en su interior el manifiesto y los recursos que se referencian en el mismo. Proporciona un formato conciso de entrega por la web, lo que permite que puedan ser utilizados para transportar los paquetes de un sistema a otro.

Para la distribución de los contenidos empaquetados, IMS CP contiene un documento fundamental llamado “Manifiesto”. Dicho documento es un fichero XML⁴ en el que se describe la estructura del paquete: sus recursos, organizaciones y los metadatos asociados a los distintos componentes.

³ **IMS Global Consortium Inc.** Organización que cuenta con miembros de organizaciones comerciales, educativas y gubernamentales dedicadas a definir y distribuir arquitecturas abiertas para actividades de educación asistido por computadora.

⁴ **XML**, Extensible Markup Language (Lenguaje de Marcas Extensible), es un metalenguaje que define la sintaxis utilizada para definir otros lenguajes de etiquetas estructurados.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<manifest identifier="CMD_7540019_M" version="1.0"
xmlns="http://www.imsproject.org/content"
xmlns:webct="http://www.webct.com/IMS">
  <metadata>
  .....
  </metadata>
  <organizations>
    <organization identifier="CMD_7540020">
      <webct:properties identifierref="CMD_7540021"/>
      <item identifier="CMD_7540022">
        <title>Temas del curso (trasparencias en pdf)</title>
      <item identifier="CMD_7540023" identifierref="CMD_7540024">
        <title>Introducción a la programación orientada a objetos</title>
      </item>
      <item identifier="CMD_7540026" identifierref="CMD_7540027">
        <title>Introducción a Java</title>
      </item>
      .....
    </organization>
  </organizations>
  <resources>
    <resource identifier="CMD_7540021" type="webctproperties">
      <file href="CMD_7540019_M/data/properties_CMD_7540020.xml"/>
    </resource>
    <resource identifier="CMD_7540024" type="webcontent">
      <file href="CMD_7540019_M/my_files/POO2004-
2005/IntroduccionPOO.pdf"/>
    </resource>
    <resource identifier="CMD_7540027" type="webcontent">
      <file href="CMD_7540019_M/my_files/POO2004-2005/introjjava.pdf"/>
    </resource>
    .....
  </resources>
</manifest>
```

Figura 1. Código de un archivo XML Manifiesto (16)

En resumen, el Manifiesto es un fichero XML que describe y organiza los contenidos de un paquete, añadiendo información adicional en forma de metadatos que pueden ser procesados y aprovechados en las tareas de catalogación de la información. Mediante el uso de IMS CP es posible encapsular en un solo fichero todos los recursos que conforman un curso y sus metadatos, o simplemente un solo recurso, dependiendo de la granularidad con que sea creado el mismo y con el fin que tenga. La estructura del Manifiesto que propone IMS CP fue empleada en la versión anterior de RHODA.

1.3.2 IEEE/Learning Object Metadata

LOM es el estándar creado por **IEEE/LTSC**⁵ que utiliza SCORM para describir los metadatos asociados a los OA, su objetivo es la creación de descripciones estructuradas de recursos educativos. Su modelo de datos especifica qué aspectos de un objeto de aprendizaje deberían ser descritos y qué vocabularios se pueden utilizar en dicha descripción. Sus elementos de metadatos se dividen en 9 categorías (17):

- **General:** describe el objeto educativo, para ello incluye campos como el identificador del OA, el título y la descripción.
- **Ciclo de vida:** describe el estado actual del objeto de aprendizaje y su historial, incluyendo información de quién ha contribuido a su evolución.
- **Metadatos:** incluye información sobre los metadatos en sí mismo.
- **Técnica:** incluye la información técnica del recurso de aprendizaje, tales como: el tamaño, la ubicación o el formato que presenta.
- **Educacional:** abarca las diferentes características pedagógicas del objeto de aprendizaje, la cual incluye campos como: tipo de recursos, ejercicios, diagramas, figuras, entre otros.
- **Derecho:** incluye los detalles sobre la propiedad intelectual del recurso, así como las condiciones de utilización y el precio en caso de tenerlo.
- **Relación:** explica el tipo de relación que tiene el recurso de aprendizaje con otros OA.
- **Anotación:** incluye comentarios sobre la utilización del OA, además de su autor y la fecha de creación.
- **Clasificación:** informa si el OA pertenece a algún tema en concreto.

Como aspecto positivo de los metadatos se puede destacar, la posibilidad que ofrecen de realizar una descripción detallada de cualquier recurso, debido a la cantidad de atributos que plantea en su estructura, lo que permite facilitar las búsquedas, clasificaciones y comprensión de los recursos.

A pesar de que LOM resulta ser un esquema de metadatos altamente reconocido y utilizado, existen otros como DublinCore y CanCore, los mismos son usados por versiones anteriores de RHODA.

⁵ **IEEE/LOM, (Institute of Electrical and Electronics Engineers/Learning Technology Standards Committee).** Asociación internacional, cuya misión es promover los procesos ingenieriles para la creación, desarrollo, integración, compartición y aplicaciones del conocimiento sobre tecnologías eléctricas y de información.

1.3.3 Modelo Referenciado de Objetos de Contenido Compartible

El Modelo Referenciado de Objetos de Contenido Compartible (SCORM), es un modelo que hace referencia a una serie de guías, especificaciones y estándares técnicos que se deben seguir para crear sistemas de aprendizaje y materiales educativos para la distribución en la web. El mismo surge como iniciativa de la Advanced Distributed Learning (ADL⁶) y especifica cómo crear cursos a partir de objetos de aprendizaje. Define mecanismos para secuenciar dichos objetos e indica qué metadatos se pueden utilizar para su identificación y localización. Además, declara cómo deben ser empaquetados y distribuidos los contenidos educativos de manera tal que se garantice su reutilización. (18)

Los detalles de la especificación se encuentran divididos en tres tópicos fundamentales, su versión más reciente se conoce como SCORM 2004 (16):

- **Modelo de Agregación de Contenido (Content Aggregation Model, CAM):** Este manual describe los distintos tipos de objetos de contenido permitidos dentro de la especificación y detalla los mecanismos que se deben seguir para su empaquetamiento, descubrimiento en repositorios y su distribución e interoperabilidad entre distintos LMS.
- **Entorno de Tiempo de Ejecución (Run-Time Environment, RTE):** El requisito de adaptabilidad e intercambio de datos entre el contenido y el LMS da lugar a contenidos educativos más complejos de lo habitual y es necesario estandarizar el proceso de ejecución de estos contenidos para garantizar la interoperabilidad entre distintos LMS. Por ello, este manual define el proceso de ejecución y los mecanismos de comunicación que tanto el LMS como el propio contenido deben emplear.
- **Secuenciamiento y Navegación (Sequencing and Navigation, SN):** Este manual define los mecanismos para que los LMS puedan concatenar las actividades educativas de modo consistente. El manual recoge los eventos que pueden ser generados por los alumnos o por el sistema y que deben ser procesados por el LMS para decidir cuál es el recurso educativo que debe ser presentado a continuación. También abarca el modelo de datos para generar y procesar estos eventos.

⁶ **ADL (Advanced Distributed Learning).** Iniciativa lanzada por el Departamento de Defensa de Estados Unidos y la oficina de Ciencias y Políticas Tecnológicas de la Casa Blanca. La misión de la ADL es proveer accesos de la más alta calidad en educación y entrenamiento, en cualquier momento.

SCORM se compone de Assets, Objetos de Contenido Compartido (SCOs por sus siglas en inglés) y la Organización de Contenidos. Los Assets son una representación electrónica de medios, textos, imágenes, sonido, páginas web, objeto de evaluación u otros datos que puedan ser procesados en un cliente web. Los SCOs, sin embargo, son una composición de uno o más Assets que representan un recurso de aprendizaje individual. La Organización de Contenidos es un mapa que representa el uso previsto del contenido a lo largo de las unidades estructuradas de enseñanza. Dentro de sus especificaciones y estándares utiliza el IMS Content Packaging para empaquetar sus contenidos y LOM para describir los metadatos asociados a ellos. (19)

1.4 Metodología de desarrollo de software

Para asegurar el éxito durante el desarrollo de un software no es suficiente contar con notaciones de modelado y herramientas, se hace necesario además el empleo de metodologías de desarrollo. Las mismas proporcionan una colección de procedimientos, técnicas, herramientas y documentos auxiliares que sirven como guía a los miembros del equipo de proyecto durante todo el proceso de desarrollo. Las metodologías de desarrollo se encuentran divididas en dos grandes grupos: las metodologías tradicionales o robustas y las metodologías ágiles, su selección depende del tipo de software que se espera construir.

Rational Unified Process o Proceso Unificado de Rational (RUP, por sus siglas en inglés) es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, tipos de organizaciones, niveles de aptitud y tamaños de proyecto. (20)

RUP es una metodología flexible y fácil de adaptarse a las necesidades de cualquier proyecto, esta brinda al equipo de desarrollo guías consistentes y personalizadas del proceso. El mismo, en conjunto con el UML, constituye la metodología estándar utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. RUP utiliza UML para definir los modelos de software y puede definirse como un modelo que es dirigido por casos de uso, centrado en la arquitectura, iterativo e incremental. (21)

1.5 Tecnologías, lenguajes y herramientas

Para la implementación del módulo de gestión de recursos educativos en el repositorio RHODA, es necesaria la utilización de un conjunto de tecnologías y herramientas, las cuales fueron definidas por especialistas del Centro de Tecnología para la Formación (FORTES), como parte del nuevo marco de trabajo. A continuación, se describen las mismas.

1.5.1 Herramienta CASE

En la actualidad, la tecnología Ingeniería de Software Asistida por Computadora (CASE por sus siglas en inglés), se define como un conjunto de programas y ayudas que dan asistencia a los analistas, ingenieros y desarrolladores de software, durante todos los pasos del ciclo de vida de desarrollo de un software. (22) Visual Paradigm es una herramienta CASE que soporta el lenguaje de modelado UML. Permite construir diagramas de diversos tipos y a partir de ellos, obtener su código correspondiente. Incluye además, la capacidad para integrarse con distintos entornos de desarrollo y gestores de base datos, facilitando la generación de los artefactos correspondientes a diagramas de clases y de base de datos. Brinda la posibilidad de exportar e importar los artefactos generados en archivos XML u otros formatos (.jpg, .png, .pdf). Posee un diseño de interfaz gráfica enfocada a facilitar la interacción y el trabajo del usuario. (23) Se utiliza Visual Paradigm en su versión 8.0 para el modelado de la propuesta de solución.

1.5.2 Lenguaje de modelado

El Lenguaje de Modelado Unificado o *Unified Modeling Language* (UML, por sus siglas en inglés), suele definirse en términos simples como: “*un lenguaje que permite modelar, construir y documentar los elementos que forman un sistema de software orientado a objetos*” (24). Es un lenguaje capaz de abstraer cualquier tipo de sistema informático, mediante diagramas que contienen toda la información relevante del sistema.

UML en su versión 2.1, permite visualizar, especificar y documentar cada una de las partes del módulo de forma estándar. Entre los beneficios que brinda se destacan los siguientes (24): ofrece un mejor soporte a la planificación y control del proyecto; permite realizar una verificación y validación del modelo realizado; facilita la documentación de las decisiones de la arquitectura del proyecto, así como especificar la estructura y comportamiento del sistema y comunicarlo a todos los integrantes.

1.5.3 Lenguaje de programación del lado del servidor

PHP

Hypertext Preprocesor (PHP), es un lenguaje de código abierto muy utilizado en el desarrollo de páginas web, el cuál puede ser insertado en el código HTML (25). Su versión 5.3 brinda las siguientes características (26):

- Soporte para diferentes gestores de bases de datos tales como: MySQL, PostgreSQL, Oracle, MS SQL, Server.
- Respaldo por una comunidad de desarrolladores, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente.
- Permite la incorporación de diferentes librerías externas, permitiendo así una mayor cantidad de opciones para desarrollar una determinada funcionalidad.

1.5.4 Lenguajes de programación del lado del cliente

HTML

El Lenguaje de Marcado de Hipertexto (HTML), es el lenguaje más utilizado para el diseño de páginas web, su versión 5.0, presenta las siguientes características (27):

- El código generado con este lenguaje es más simple, lo que permite hacer páginas más ligeras que se cargan rápidamente favoreciendo la usabilidad y la indexación en buscadores.
- Ofrece una compatibilidad mayor con los navegadores de dispositivos móviles.
- Posibilita la inserción de vídeos y audio de forma directa.
- Permite realizar diseños adaptables a distintos dispositivos (web, tablets y móviles).

Para el desarrollo de la propuesta de solución se utiliza HTML5.

JavaScript

JavaScript es un lenguaje de programación que se utiliza principalmente para crear páginas web dinámicas. Una página web dinámica es aquella que incorpora efectos tales como textos que aparecen y desaparecen, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de aviso al usuario. Técnicamente, JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlo (28). Para el desarrollo de la propuesta de solución se utiliza la versión 1.3, la cual ofrece una mejor práctica de validación de datos del lado del cliente, requiere poca memoria RAM y es soportado por los navegadores más utilizados en la actualidad.

CSS

El lenguaje Hoja de Estilo en Cascada o Cascading Style Sheets (CSS, por sus siglas en inglés), es un lenguaje del lado del cliente que define la presentación de los documentos HTML. Algunos de los beneficios de versión 3 son (29):

- Establece un control de la presentación de diferentes documentos desde una única hoja de estilo.
- Mejora la accesibilidad del documento y reduce la complejidad de su mantenimiento.
- Separación del contenido de la presentación, lo que trae consigo la producción de códigos más organizados.
- Nuevas propiedades especialmente en cuanto al aspecto gráfico.

Para el desarrollo de la propuesta de solución se utiliza CSS3.

1.5.5 Framework de desarrollo

Symfony

Symfony2, es la versión más reciente del *framework* Symfony para el desarrollo de aplicaciones utilizando el lenguaje PHP. Es un *framework* completo, diseñado para optimizar el desarrollo de las aplicaciones web, basado en el patrón Modelo Vista Controlador. Automatiza las tareas más comunes, permitiendo al desarrollador dedicarse a tiempo completo a las funcionalidades específicas de la aplicación web. Posee facilidad de instalación y configuración en plataformas Unix y Windows. Es independiente del Sistema Gestor de Base de Datos (SGBD) que se utilice. Permite la internacionalización para la traducción del texto de la interfaz, los datos y el contenido de localización. (30)

Su versión 2.3.7, contiene diversas características y ventajas con respecto a Symfony 1.4, entre las que se encuentran (30):

- Encapsula operaciones complejas en instrucciones sencillas.
- Usa el paradigma de Programación Orientada a Objetos (POO).
- Establece oficialmente la técnica para mapear los objetos relaciones (ORM) Doctrine.
- Soporta en la capa de presentación el uso de los framework jQuery y Bootstrap.

JQuery

El *framework* jQuery contiene una serie de funciones y métodos para el lenguaje JavaScript. Implementa una serie de clases de programación orientadas a objetos que permiten programar sin preocuparse del navegador que utilice el usuario. El *framework* jQuery en su versión 1.10 facilita la creación de interfaces de usuario, efectos dinámicos y aplicaciones que hagan uso de otras tecnologías como Ajax⁷. Además, su licencia implica el uso en cualquier plataforma personal o comercial. (31)

Para el desarrollo de la propuesta de solución se utiliza jQuery en su versión 1.10.

Bootstrap

Bootstrap es un *framework* de CSS diseñado para simplificar el proceso de creación de diseños web, además es compatible con los navegadores actuales. Para el desarrollo de la propuesta de solución se utiliza la versión 3.0. La misma ofrece una serie de plantillas CSS y de ficheros JavaScript, que permiten conseguir (32):

- Un diseño que puede ser visualizado de forma correcta en distintos dispositivos a distintas escalas y resoluciones.
- Una mejor integración con librerías de otros *framework* como jQuery.
- Un diseño sólido con el uso de lenguajes como CSS3 y HTML5.

1.5.6 Entorno de desarrollo

Un entorno de desarrollo integrado, en inglés *Integrated Development Environment* (IDE), se refiere a un programa o interfaz visual de programación que se utiliza para interactuar con el compilador y/o depurador de un lenguaje de programación determinado. Los IDE proveen un marco de trabajo enfocado a facilitar la interacción y el trabajo con lenguajes de programación tales como: C++, Python, Java, C#, Delphi, Visual Basic.

NetBeans IDE

⁷ Ajax: acrónimo de Asynchronous JavaScript And XML. Técnica de desarrollo web para crear aplicaciones interactivas.

Netbeans es una herramienta que permite a los desarrolladores de software escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero puede ser utilizado en cualquier otro lenguaje de programación. Posee un número importante de módulos que amplían y facilitan su utilización. Es un producto libre, gratuito y sin restricciones de uso.

La versión a utilizar en la presente investigación es 7.3.1, que ofrece un rendimiento superior para desarrolladores de aplicaciones web, proporcionando editores y herramientas integrales con otros lenguajes como HTML, JavaScript y CSS. Posee un editor y depurador de código php e incluye además, la capacidad de integración con el *framework* Symfony y con sistemas de control de versiones.

1.5.7 Sistema gestor de base de datos

Un SGBD se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Dicho sistema ayuda a realizar las siguientes acciones: definición y mantenimiento de la integridad de los datos, control de la seguridad, privacidad y manipulación de los mismos (33). Entre los principales SGBD se encuentran: PostgreSQL y MYSQL.

PostgreSQL

Es un SGBD objeto-relacional, distribuido bajo la licencia PostgreSQL, que solo requiere que el código fuente mantenga su información de derecho de autor (*copyright*) y licencia. Es una herramienta construida sobre código abierto. Utiliza un modelo cliente/servidor y usa tecnología multiprocesos para garantizar la estabilidad del sistema, por lo que un fallo en uno de los procesos no afecta el resto. (34)

En la presente investigación se utiliza PostgreSQL en su versión 9.1, permitiendo a los autores una mejor integridad referencial de los datos, múltiples métodos de autenticación y acceso encriptado vía SSL⁸.

1.5.8 Servidor web

Un servidor web es un programa que sirve para atender y responder a las diferentes peticiones de los navegadores, proporcionando los recursos solicitados por los usuarios, usando el protocolo *Hypertext Transfer Protocol* (HTTP) o el protocolo HTTPS (versión cifrada y autenticada).

⁸**SSL Secure Sockets Layer.** Protocolo diseñado para transmitir información de manera segura entre aplicaciones. Disponible en <http://www.digicert.com/es/ssl.htm>

Apache

Apache es un servidor web desarrollado por la Fundación de Software Apache, cuyo objetivo es servir o suministrar páginas web a clientes y/o navegadores web que las solicitan. Es una plataforma de servidor web de código abierto, lo que permite su ejecución en sistemas operativos como Windows, Macintosh, y Linux. Es altamente configurable y utiliza el protocolo HTTP para la transferencia de páginas. Las características antes presentadas hacen que sea centrado en la escalabilidad, seguridad y rendimiento (35). Para la propuesta de solución se utiliza el servidor web Apache la versión 2.2.

1.6 Arquitectura de software

La arquitectura de software permite la comunicación entre todas las partes implicadas en el desarrollo del sistema. Destaca las decisiones iniciales relacionadas con el diseño, lo que permite el éxito final del sistema como una entidad operacional. Además, proporciona un modelo comprensible de cómo está estructurado el sistema y las relaciones entre los componentes. (36)

Para el desarrollo de la propuesta de solución se utiliza el patrón arquitectónico:

Modelo-Vista-Controlador (MVC)

Teniendo en cuenta que el patrón arquitecto MVC, es el patrón utilizado por el *framework* Symfony para el desarrollo de cualquier sistema, se decide definir la arquitectura del módulo de acuerdo con las especificaciones de dicho *framework*. El patrón MVC, separa la lógica de negocio (el modelo) y la presentación (vista), permitiendo que el desarrollo y mantenimiento del módulo sea rápido y sencillo. El MVC está formado por tres niveles (37):

- Modelo: representa la información con la que trabaja la aplicación, es decir la lógica del negocio.
- Vista: transforma el modelo en una página web que permite al usuario interactuar con ella.
- Controlador: se encarga de procesar las interacciones del usuario y realizar los cambios apropiados en el modelo o en las vista.

Conclusiones parciales

El estudio de la muestra de repositorios de recursos educativos abordado en este capítulo permitió identificar las funcionalidades básicas que debe presentar un repositorio para realizar una completa gestión de los recursos que almacena. El análisis de los estándares para el empaquetamiento y descripción de los contenidos utilizados en versiones anteriores del repositorio RHODA, permitió ratificar la necesidad de su incorporación en la propuesta de solución a fin de garantizar la homogenización de la

estructura de los mismos. El estudio de las tecnologías, herramientas y metodología de desarrollo de software, permitió la adquisición de conocimientos y experiencias en el trabajo con las mismas.

Capítulo 2. Propuesta de solución

Introducción

La comprensión de los requisitos de un problema, se encuentra entre las tareas más difíciles que enfrenta hoy en día cualquier ingeniero de software, debido a que en realidad se está realizando la caracterización que permitirá obtener la primera vista conceptual del mismo. Las características de un sistema, son los elementos claves para el desarrollo de los artefactos ingenieriles que permiten obtener una vista estática del sistema, el modelado de las mismas, permite ofrecer posteriormente una entrada coherente al análisis y diseño y que finalmente será transformada en código fuente. De esta forma al plantear las características de la propuesta de solución, se están sentando las primeras bases de la arquitectura de la futura solución.

2.1 Modelo del dominio

El modelo del dominio captura, comprende y describe los objetos más importantes dentro del contexto de un sistema (20). Se describe mediante diagramas de UML, especialmente mediante diagramas de clases. Los objetos del dominio o clases pueden obtenerse a partir de una especificación de requisitos o mediante la entrevista con los expertos del tema. Las clases que lo componen suelen aparecer en tres formas típicas (20):

- Objetos del negocio que representan los elementos que se manipulan en el negocio.
- Objetos del mundo real y conceptos de los que el sistema debe hacer un seguimiento.
- Sucesos que ocurrirán o que han ocurrido.

El modelo de dominio que a continuación se describe constituye una representación gráfica de los principales conceptos que forman parte del módulo para la gestión de recursos educativos en RHODA.

Definición de las clases

RHODA: repositorio desarrollado en el centro FORTES que almacena recursos educativos.

Usuario: persona registrada en el sistema.

Recurso educativo: recurso compuesto por contenidos educativos que pueden ser cursos completos, materiales para cursos, materiales multimedia, objetos de aprendizaje y cualquier otro recurso con fin educativo con una alta pertinencia y certificación de calidad.

Autor: persona registrada en el sistema que posee los permisos para subir recursos al repositorio, a través de una herramienta de Autor y editar las propiedades del mismo.

Administrador: persona registrada en el sistema que cuenta con todos los privilegios, es el encargado de todas las configuraciones del repositorio.

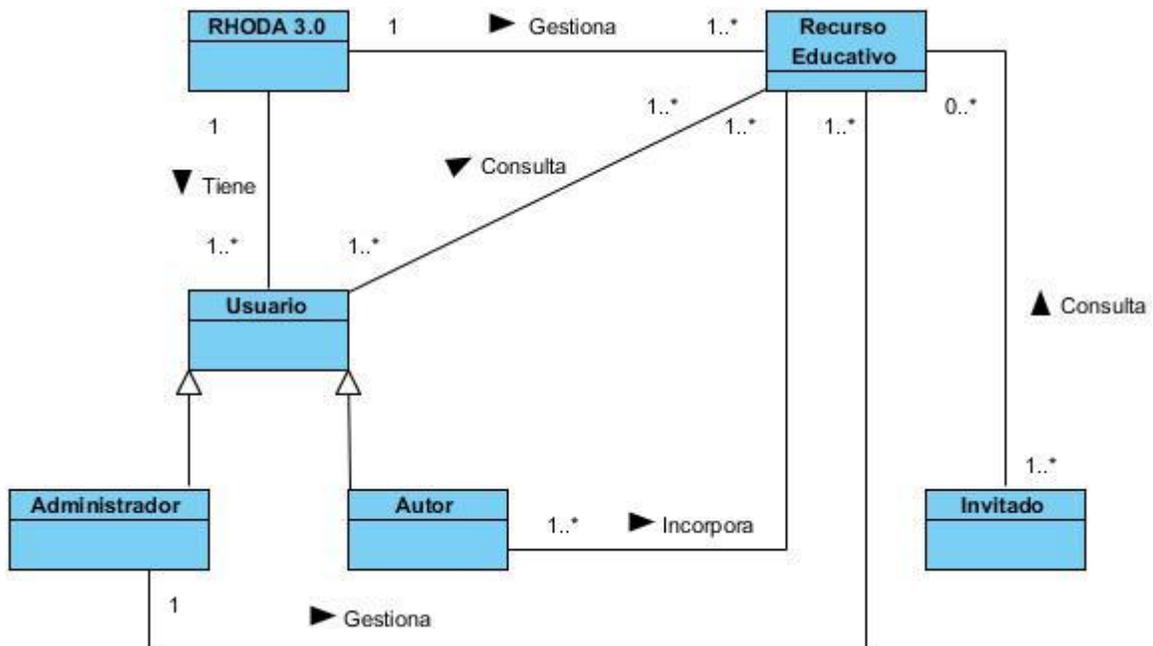


Figura 2. Modelo de dominio

2.2 Requisitos funcionales y no funcionales

El flujo de trabajo Requerimientos, constituye uno de los flujos más importante que propone la metodología RUP, pues en el mismo se capturan los requisitos del sistema. Los mismos tienen como objetivo fundamental guiar el desarrollo hacia un sistema correcto.

2.2.1 Requisitos funcionales

Los requisitos funcionales representan las funcionalidades del sistema y se modelan mediante diagramas de casos de uso. Para la propuesta de solución se definen los siguientes requisitos funcionales:

- **RF1. Listar los recursos educativos más recientes.** Permite a los usuarios del sistema incluyendo al invitado, listar los cinco últimos recursos que han sido publicados en el repositorio.
- **RF2. Listar los recursos educativos más solicitados.** Permite a los usuarios del sistema incluyendo al invitado, listar los cinco recursos más solicitados por los usuarios del repositorio.
- **RF3. Listar los recursos educativos más visualizados.** Permite a los usuarios del sistema incluyendo al invitado, listar los cinco recursos que han sido más visualizados.
- **RF4. Listar los recursos educativos más descargados.** Permite a los usuarios del sistema incluyendo al invitado, listar los cinco recursos que han sido más descargados.
- **RF5. Ver ficha de un recurso educativo.** Permite a los usuarios del sistema incluyendo al invitado, ver la ficha de un recurso educativo.
- **RF6. Ver comentario de un recurso educativo.** Permite a los usuarios del sistema incluyendo al invitado ver los comentarios de un recurso educativo.
- **RF7. Visualizar recurso educativo.** Permite a los usuarios del sistema incluyendo al invitado, visualizar el contenido de un recurso educativo.
- **RF8. Descargar recurso educativo.** Permite a los usuarios del sistema incluyendo al invitado, descargar un recurso educativo.
- **RF9. Importar recurso educativo.** Permite al autor y al administrador importar un recurso educativo al repositorio.
- **RF10. Editar las propiedades de un recurso educativo.** Permite al autor de un recurso y al administrador del sistema, editar las propiedades de un recurso.
- **RF11. Eliminar recurso educativo.** Permite al administrador eliminar un recurso del repositorio.
- **RF12. Reemplazar recurso educativo.** Permite al autor de un recurso o administrador del sistema reemplazar un recurso que se encuentre en el repositorio.
- **RF13. Exportar recurso educativo.** Permite a los usuarios del sistema incluyendo al invitado, exportar los recursos a formatos SCORM 1.2 y SCORM 2004 usando el lenguaje de catalogación LOM.

- **RF14. Adicionar comentario a un recurso educativo.** Permite a los usuarios introducir su valoración sobre un determinado recurso.
- **RF15. Editar comentarios de un recurso educativo.** Permite a los usuarios editar sus comentarios sobre un recurso.
- **RF16. Eliminar comentarios de un recurso educativo.** Permite al administrador eliminar un comentario realizado a un recurso.
- **RF17. Filtrar comentario de un recurso educativo.** Permite a los usuarios filtrar los comentarios de un recurso por un rango de fecha.
- **RF18. Denunciar comentario de un recurso educativo.** Permite a los usuarios denunciar un determinado comentario que se ha realizado a un recurso.
- **RF19. Responder comentario de un recurso educativo.** Permite a los usuarios introducir la valoración personal de un determinado comentario a un recurso educativo.
- **RF20. Recomendar recurso educativo a un usuario.** Permite a los usuarios recomendar un recurso a otro usuario.
- **RF21. Denunciar un recurso educativo.** Permite a los usuarios denunciar un determinado recurso educativo.
- **RF22. Calificar un recurso educativo.** Permite a los usuarios otorgarle una calificación a un recurso educativo.
- **RF23. Realizar sugerencias al autor de un recurso educativo.** Permite a los usuarios realizar sugerencias al autor de un determinado recurso educativo.
- **RF24. Listar las sugerencias de todos los recursos educativos.** Permite a los usuarios ver las sugerencias de todos recursos educativos.
- **RF25. Listar sugerencias de un recurso educativo.** Permite a los usuarios ver las sugerencias que ha recibido un determinado recurso educativo.
- **RF26. Ver sugerencias de un recurso educativo.** Permite a los usuarios ver las sugerencias que se han realizado a un recurso educativo.

- **RF27. Eliminar sugerencia de un recurso educativo.** Permite al administrador eliminar las sugerencias que se han realizado a un recurso.
- **RF28. Compartir información de los recursos educativos en la red social Facebook.** Permite a los usuarios compartir en la red social facebook información relacionada a un determinado recurso.
- **RF29. Compartir información de los recursos educativos en la red social Twitter.** Permite a los usuarios compartir en la red social twitter información relacionada a un determinado recurso.
- **RF30. Adicionar recurso educativo a favoritos.** Permite a los usuarios adicionar recursos educativos a la categoría favoritos.
- **RF31. Listar los recursos educativos favoritos.** Permite a los usuarios listar los recursos educativos de la categoría favoritos.
- **RF32. Listar publicaciones.** Permite al autor listar sus recursos educativos en el repositorio.

2.2.2 Requisitos no funcionales

Los requisitos no funcionales son aquellos que no se refieren a las funcionalidades que proporciona el sistema, sino a las propiedades como restricciones del entorno o de la implementación, rendimiento, dependencias y soportes entre otras.

➤ Usabilidad

RnF 1 Ambiente

Se debe contar con un servidor de Bases de Datos Relacional PostgreSQL y un Servidor de Aplicaciones web, cada uno con más de 2 GB de memoria RAM y 160 GB de Disco Duro en el caso del Servidor de Bases de Datos. El acceso a la aplicación se realizará desde varias estaciones clientes, las que deberán tener más de 512 MB de memoria RAM y un navegador actual. El tiempo de respuesta de la aplicación no debe exceder los 5 segundos para cada petición realizada.

➤ Confiabilidad

RnF 2 Respuesta ante inactividad del usuario

El sistema debe cerrar la sesión de un usuario luego de que el mismo se haya mantenido inactivo durante 15 minutos.

RnF 3 Respuesta ante ocurrencia de una excepción

El sistema debe ser capaz de mostrar un mensaje de error cuando un usuario intente acceder a un recurso educativo, indicando que el mismo ha sido eliminado.

- **Eficiencia**

RnF 5 Rendimiento

La cantidad de datos que son transmitidos por segundos es de 500 KB.

- **Soporte**

RnF 6 Estándar de codificación

El estándar de codificación utilizado es el definido en el marco de trabajo Xalix.

- **Restricciones de diseño**

RnF 7 Lenguaje de programación

El lenguaje de programación utilizado es PHP 5.3.

RnF 8 Entorno de Desarrollo Integrado

El Entorno de Desarrollo Integrado utilizado es Netbeans 7.3.1.

RnF 9 Framework de desarrollo

El *framework* de desarrollo utilizado es Symfony 2.3.7.

RnF 10 Estilo arquitectónico

El estilo arquitectónico utilizado es Modelo Vista Controlador.

RnF 11 Biblioteca JavaScript

La biblioteca JavaScript utilizada es jQuery v1.10.

RnF 12 Framework CSS

El *framework* CSS utilizado para el diseño de las interfaces gráficas es Bootstrap v 3.0.

- **Requisitos de licencia**

RnF 13 Licencia

El repositorio está registrado bajo la Licencia Pública General (GPL) versión 3.0.

➤ **Estándar aplicable**

RnF 14 LOM

El repositorio utiliza la especificación LOM para la descripción de los recursos educativos incluyendo sus nueve categorías.

RnF 15 SCORM

El repositorio utiliza el estándar SCORM y dentro de este IMS CP para la estandarización y empaquetamiento de los recursos educativos.

2.3 Identificación de los actores y casos de uso del sistema

Los actores representan personas u otro software que interactúan con el sistema. Estos se comunican con el mismo a través del envío y recepción de mensajes según este lleve a cabo los casos de uso. (20)

Con el objetivo de especificar privilegios, nivel de acceso a la información y funciones en el módulo de gestión de recursos, se identifican los siguientes actores:

Tabla 1. Descripción de los actores del sistema

Actores	Descripción
Invitado	Persona que no posee un nombre de usuario en el sistema y que por tanto solo tendrá acceso a sus funciones básicas.
Usuario	Persona registrada en el sistema.
Autor	Persona registrada en el sistema que posee los privilegios para subir recursos al repositorio.
Administrador	Persona registrada en el sistema que cuenta con todos los privilegios.

2.3.1 Casos de uso del sistema

Un caso de uso especifica una secuencia de acciones, incluyendo variantes, que el sistema puede llevar a cabo y que producen un resultado observable de valor para un actor concreto. (20)

A continuación se enuncian los casos de uso (CU) del sistema para la propuesta de solución:

- **CU_1:** Listar recursos educativos por categoría.
- **CU_2:** Importar recurso educativo.
- **CU_3:** Exportar recurso educativo.
- **CU_4:** Descargar recurso educativo.
- **CU_5:** Ver comentarios de un recurso educativo.
- **CU_6:** Ver ficha de un recurso educativo.
- **CU_7:** Visualizar recurso educativo.
- **CU_8:** Eliminar recurso educativo.
- **CU_9:** Editar las propiedades de un recurso educativo.
- **CU_10:** Reemplazar recurso educativo.
- **CU_11:** Adicionar comentario a un recurso educativo.
- **CU_12:** Editar comentario de un recurso educativo.
- **CU_13:** Responder comentario de un recurso educativo.
- **CU_14:** Filtrar comentario de un recurso educativo.
- **CU_15:** Denunciar comentario de un recurso educativo.
- **CU_16:** Eliminar comentario de un recurso educativo.
- **CU_17:** Calificar recurso educativo.
- **CU_18:** Denunciar recurso educativo.
- **CU_19:** Recomendar recurso educativo a otro usuario.
- **CU_20:** Adicionar sugerencias al autor de un recurso educativo.
- **CU_21:** Listar sugerencias de un recurso educativo.
- **CU_22:** Listar sugerencias de todos recursos educativos.
- **CU_23:** Ver sugerencia de un recurso educativo.
- **CU_24:** Eliminar sugerencias al autor de un recurso educativo.
- **CU_25:** Compartir recurso educativo en redes sociales.
- **CU_26:** Adicionar recurso educativo a favoritos.
- **CU_27:** Listar recursos educativos favoritos.
- **CU_28:** Listar publicaciones.

Patrón de casos de uso utilizado

La utilización de patrones de CU ha demostrado ser útil en el desarrollo de modelos de CU mantenibles y reutilizables. Estos patrones se centran en diseños y técnica de alta calidad y no en cómo representar casos de usos específicos (38). Una vez obtenidos los CU se aplicó el patrón con el propósito de crear un modelo más sencillo y estructurado. El patrón aplicado se menciona a continuación.

- **Extensión o Inclusión: Modelo de extensión**

Consiste en dos casos de uso y una relación extendida entre ellos. El caso de uso extensión es concreto y puede ser inicializado por sí mismo, así como ampliar el caso de uso base. Es aplicable cuando un flujo puede extender el flujo de otro caso de uso.

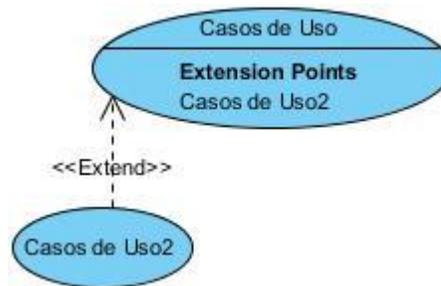


Figura 3. Modelo de extensión

2.3.2 Diagrama de casos de uso del sistema

El diagrama de casos de uso del sistema representa gráficamente los procesos y su interacción con los actores del sistema. En la siguiente figura se muestra el diagrama de CU correspondiente a la propuesta de solución.



Figura 4. Diagrama de CU del sistema

2.3.3 Descripciones de los CU

Las descripciones de los CU permiten documentar en un lenguaje cliente/usuario una lista de pasos que debe seguir el actor para interactuar con el sistema. Posibilitando además realizar un análisis y a partir de este poder realizar los diagramas de clases del análisis, estructurar los del diseño. En esta sección se presenta la descripción de los CU más significativos del módulo, las restantes se podrán encontrar en el **Anexo I**.

Tabla 2. Descripción del CU: Importar recurso educativo

Objetivo	Importar un recurso educativo al repositorio.	
Actores	Autor (inicia).	
Resumen	El caso de uso inicia una vez que el autor desea subir al repositorio un recurso educativo y finaliza cuando este es subido al mismo.	
Complejidad	Alta.	
Prioridad	Crítico.	
Precondiciones	El recurso educativo no puede exceder el tamaño definido por el administrador del sistema.	
Postcondiciones	El recurso educativo es almacenado en el repositorio.	
Flujo de eventos		
Flujo básico: Importar recurso educativo		
	Actor	Sistema
1.	Selecciona la opción “Importar recursos educativos” que se encuentra en la lista desplegable <i>Recursos Educativos</i> .	<p>1.1 Muestra la interfaz que permite importar un recurso educativo al repositorio, la misma contiene los siguientes campos:</p> <ul style="list-style-type: none"> • Categoría: cuadro de selección que permite elegir la categoría a la que pertenecerá el recurso educativo: <ul style="list-style-type: none"> - Video - Imagen - Documento - Objeto de Aprendizaje. • Colección: cuadro de selección que permite elegir la colección a la que pertenecerá el recurso educativo. • Localización: campo de tipo file que permite subir un archivo. <p>Además muestra las siguientes opciones:</p> <ul style="list-style-type: none"> • Subir: botón que permite almacenar el recurso educativo al repositorio. • Cancelar: botón que permite regresar a la vista de inicio.
2.	Selecciona la categoría.	2.1. Muestra la interfaz que permite introducir los datos del

		<p>recurso, si no es un Objeto de Aprendizaje, la misma contiene los siguientes campos de textos obligatorios :</p> <ul style="list-style-type: none"> • Título: campo de texto que permite insertar el título correspondiente al recurso educativo. • Descripción: campo de texto que permite insertar la descripción correspondiente al recurso educativo. • Palabras claves: campo de texto que permite insertar las palabras claves correspondiente al recurso educativo.
3.	Introduce los datos y selecciona la opción <i>Subir</i> .	<p>3.1 Valida los campos.</p> <p>3.2 Almacena el recurso educativo en el repositorio.</p> <p>3.3 Muestra el mensaje de confirmación: “<i>El recurso se ha importado correctamente</i>”.</p>
4.		4.1 Termina el caso de uso.
Flujos alternos		
2.a “Selecciona la categoría Objeto de Aprendizaje”		
	Actor	Sistema
2.a	Selecciona la opción <i>Subir</i> .	4.1a Retorna al punto 3.1 del flujo básico.
3.1 b “Existen campos vacíos ”		
	Actor	Sistema
3.b		<p>3.1b Resalta de color rojo los campos vacíos.</p> <p>3.2b Retorna al punto 2.1 del flujo básico.</p>
3.1 c “Formato no válido ”		
	Actor	Sistema
3.c		<p>3.1c Muestra el mensaje: “<i>Formato no válido</i>”.</p> <p>3.2c Retorna al punto 2.1 del flujo básico.</p>
1.1a “Opción Cancelar ”		
	Actor	Sistema
1.a	Selecciona la opción <i>Cancelar</i> .	<p>1.1a Retorna a la vista de inicio.</p> <p>1.2a Termina el caso de uso.</p>

Requisitos funcionales	RF9
-------------------------------	-----

Tabla 3. Descripción del CU: Descargar recurso educativo

Objetivo	Descargar un recurso educativo que se encuentre en el repositorio.	
Actores	Invitado (inicia).	
Resumen	El caso de uso se inicia cuando el invitado selecciona la opción <i>Descargar</i> en la ficha de un recurso educativo y finaliza una vez que se produce la descarga.	
Complejidad	Alta.	
Prioridad	Crítico.	
Precondiciones	Debe haberse ejecutado el CU: <i>Ver ficha de un recurso educativo</i> .	
Postcondiciones	El recurso educativo es descargado.	
Flujo de eventos		
Flujo básico Descargar recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Descargar</i> .	1.1 Muestra la interfaz que permite seleccionar el lugar donde desea descargar el recurso educativo. 1.2 Descarga el recurso educativo en la dirección seleccionada en formato .Zip.
2.		2.1 Termina el caso de uso.
Requisitos funcionales	RF6	

Tabla 4. Descripción del CU: Recomendar un recurso educativo a otro usuario

Objetivo	Recomendar un recurso educativo a otro usuario.
Actores	Usuario (inicia).
Resumen	El caso de uso inicia una vez que el usuario accede a la opción <i>Recomendar</i> . El sistema le permitirá al usuario enviar una recomendación de un recurso educativo a otro usuario.
Complejidad	Media.

Prioridad	Secundario.	
Precondiciones	El usuario esté autenticado y debe haberse ejecutado el CU: <i>Ver ficha de un recurso educativo.</i>	
Postcondiciones	Se envió el correo al destinatario.	
Flujo de eventos		
Flujo básico: Recomendar recurso educativo a otro usuario.		
	Actor	Sistema
1.	Selecciona la opción <i>Recomendar</i> .	<p>1.1 Muestra la interfaz que permite recomendar un recurso educativo, la misma contiene los siguientes campos:</p> <ul style="list-style-type: none"> • Correo electrónico: campo de texto que permite insertar una dirección de correo electrónico. • Contenido: campo de texto que permite insertar un mensaje. <p>Además muestra las siguientes opciones:</p> <ul style="list-style-type: none"> • Enviar: botón que permite enviar el correo al destinatario. • Cancelar: botón que permite retornar a la ficha del recurso educativo.
2.	Introduce los datos necesarios y selecciona la opción <i>Enviar</i> .	<p>2.1. Valida los campos.</p> <p>2.2. Envía la recomendación al usuario destinatario.</p> <p>2.3. Muestra el mensaje: <i>“La recomendación se ha enviado correctamente”</i>.</p>
3.		3.1 Termina el caso de uso.
Flujos alternos		
1.1a “Cancelar ”		
	Actor	Sistema
1.a	Selecciona la opción <i>Cancelar</i> .	1.1a Retorna a la vista de la ficha del recurso

		educativo. (Ver CU: <i>Ver ficha del recurso educativo</i>).
2.1a “ Existen campos vacíos”		
	Actor	Sistema
2.a		2.1a Muestra el mensaje: <i>“Campo vacíos”</i> . 2.2a Retorna el punto 2.1 de flujo básico.
2.1b “Formato de correo electrónico incorrecto ”		
	Actor	Sistema
2.b		2.1b Muestra el mensaje: <i>“Introduzca una dirección de correo válida ”</i> 2.2b Retorna el punto 2.1 de flujo básico.
Requisitos funcionales	RF20	

Conclusiones parciales

La identificación de los requisitos tanto funcionales como no funcionales, permitió elaborar un modelo del sistema que sirvió de acuerdo entre los desarrolladores y el cliente, constituyendo además la entrada fundamental para las fases posteriores del desarrollo. Las descripciones de los CU permitieron conocer el funcionamiento general del sistema, estableciendo para ellos un conjunto de pasos lógicos, flujos de información y respuestas del sistema al usuario, logrando obtener una comprensión general del negocio.

Capítulo 3. Análisis y diseño

Introducción

El análisis tiene como propósito fundamental conseguir una comprensión más precisa de los requisitos y una descripción de los mismos que sea más fácil de mantener y que ayude a estructurar el sistema completo, incluyendo la arquitectura (20). El diseño permite analizar los requerimientos refinándolo y estructurándolos con el objetivo de dar soporte a todos los requisitos funcionales y no funcionales que se incorporan en el mismo (20). La concepción de los modelos de análisis y diseño es una de las tareas más complejas e importantes dentro del desarrollo de productos de software, cuando se requiere que los sistemas puedan ser interpretados por terceros ajenos al negocio, lo cuál es el objetivo fundamental del presente capítulo.

3.1 Modelo de análisis

El modelo de análisis es un modelo de objetos conceptuales que analiza los requerimientos mediante su refinación y estructuración. Con la creación del mismo, se persiguen tres objetivos específicos: describir lo que requiere el cliente; establecer una base para la creación de un diseño de software y definir un conjunto de requisitos que puedan validarse una vez construido el mismo. (20)

3.1.1 Diagramas de clases del análisis

Las clases del análisis constituyen el artefacto fundamental del modelo de análisis (36). Los diagramas de clases del análisis (DCA) se utilizan para mostrar las clases y sus relaciones, además, presentar los subsistemas e interfaces. Una clase de análisis representa una abstracción de una o varias clases y/o subsistemas del diseño de un sistema, las mismas se representan mediante los siguientes estereotipos estandarizados en UML lo que permite a los desarrolladores, distinguir el ámbito de las diferentes clases. (20)

Clases de Interfaz (CI): se utilizan para modelar las interacciones entre el sistema y sus actores. Representan abstracciones de ventanas, formularios, paneles, interfaces de comunicaciones, de impresoras, sensores y terminales.

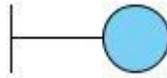


Figura 5. Estereotipo que representa la CI

Clases de Entidad (CE): se utilizan para modelar la información que posee larga vida y que es a menudo persistente. Suelen mostrar una estructura de datos lógica y contribuyen a comprender de qué información depende el sistema.

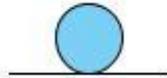


Figura 6. Estereotipo que representa la CE

Clases de Control (CC): representan coordinación, secuencia, transacciones y control de otros objetos y se usan con frecuencia para encapsular el control de un caso de uso en concreto, además, el manejo y coordinación de las acciones a otros objetos es decir, objetos de interfaz y de entidad.



Figura 7. Estereotipo que representa la CC

A continuación, se muestran los DCA correspondientes a los CU: Importar recurso educativo, Descargar recurso educativo y Recomendar recurso educativo, los restantes DCA se pueden consultar en el **Anexo II**.

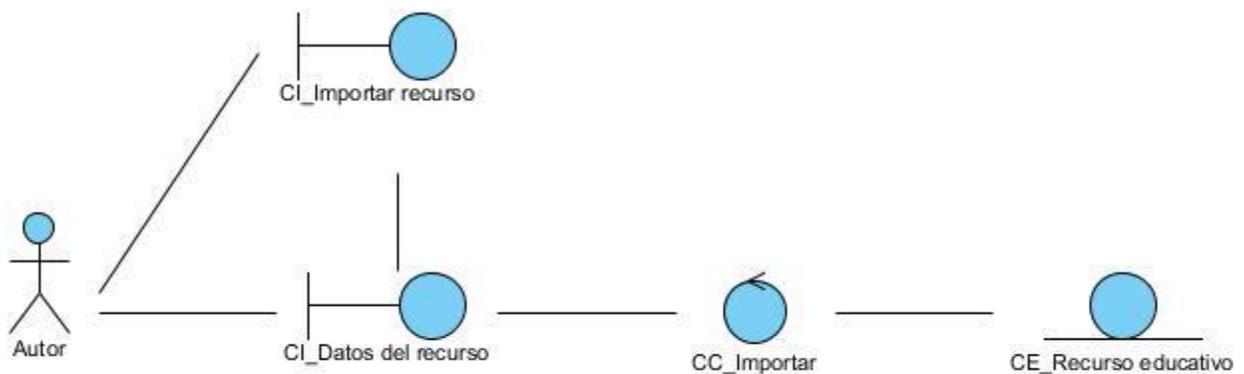


Figura 8. DCA del CU: Importar recurso educativo

En la figura anterior se presenta el diagrama de clases de análisis del CU: Importar recurso educativo, la cual ofrece una idea de cómo llevar a cabo el proceso de importar recursos educativos en el repositorio. Cuenta con la clase interfaz Importar recurso, que permite seleccionar la acción en el sistema. Dicha clase se relaciona con la clase interfaz Datos del recurso, lo que posibilita que el autor de un recurso pueda insertar los datos asociados al mismo. Esta interfaz se relaciona con la clase controladora Importar, quien maneja los eventos y especificaciones del CU controlando las acciones del mismo. Esta a su vez se comunica con la clase entidad Recurso educativo que posee la información relacionada a los recursos educativos.

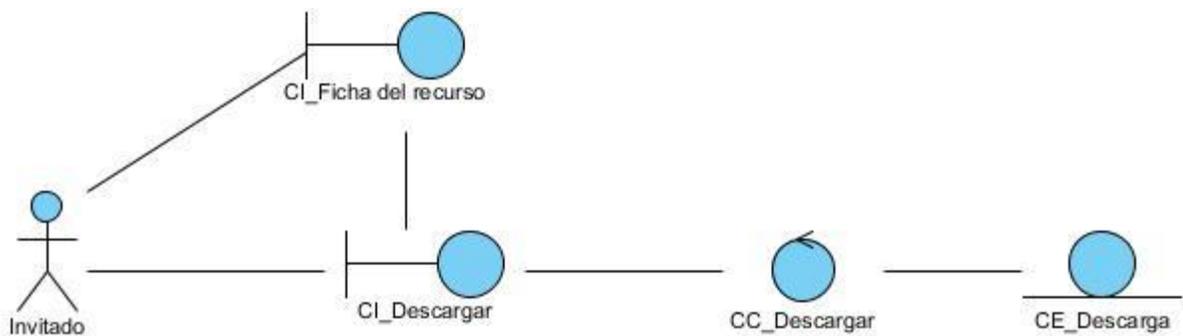


Figura 9. DCA del CU: Descargar recurso educativo

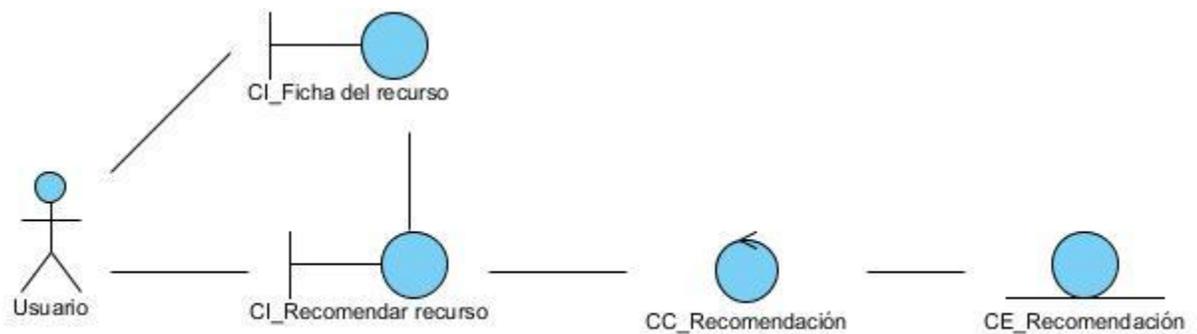


Figura 10. DCA del CU: Recomendar recurso educativo

3.1.2 Diagramas de colaboración

Los diagramas de colaboración (DC) son los artefactos que se generan durante el flujo de trabajo Análisis y Diseño, los mismos permiten mostrar las interacciones entre los objetos del análisis, creando enlaces y añadiendo mensajes entre ellos. A continuación se muestran los DC correspondientes a los CU: Importar recurso educativo, Descargar recurso educativo y Recomendar recurso educativo, el resto de los diagramas se encuentran en el **Anexo III**.

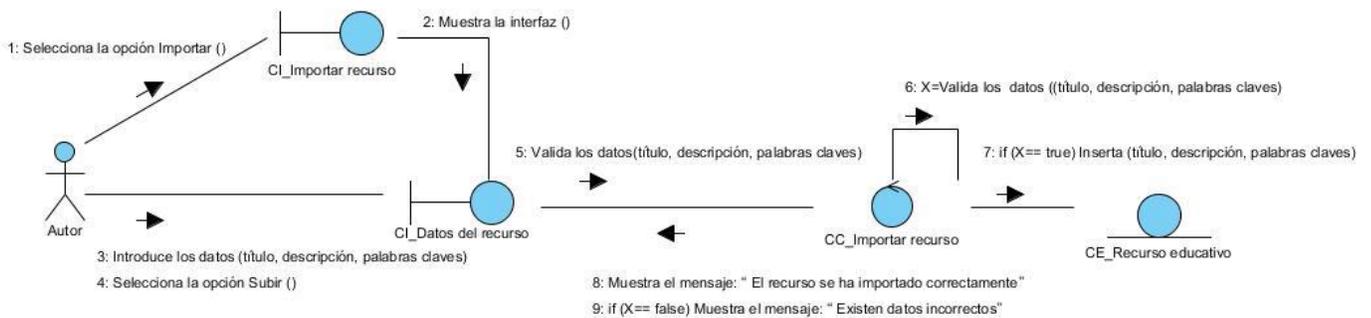


Figura 11. DC del CU: Importar recurso educativo

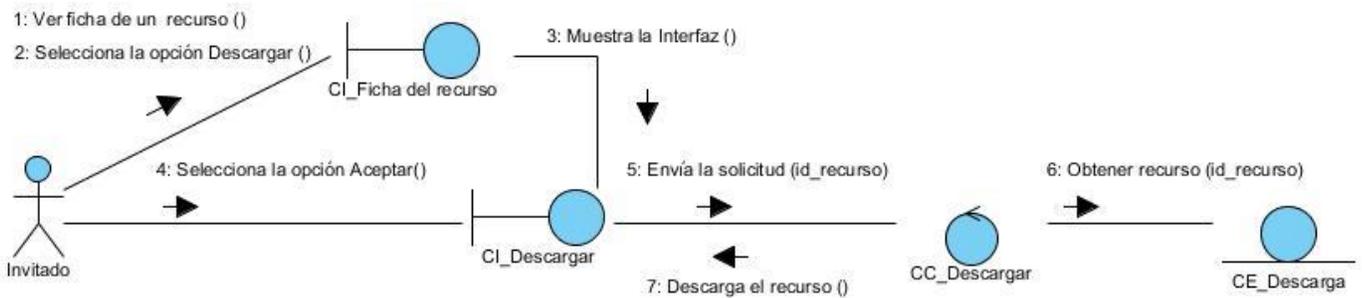


Figura 12. DC del CU: Descargar recurso educativo

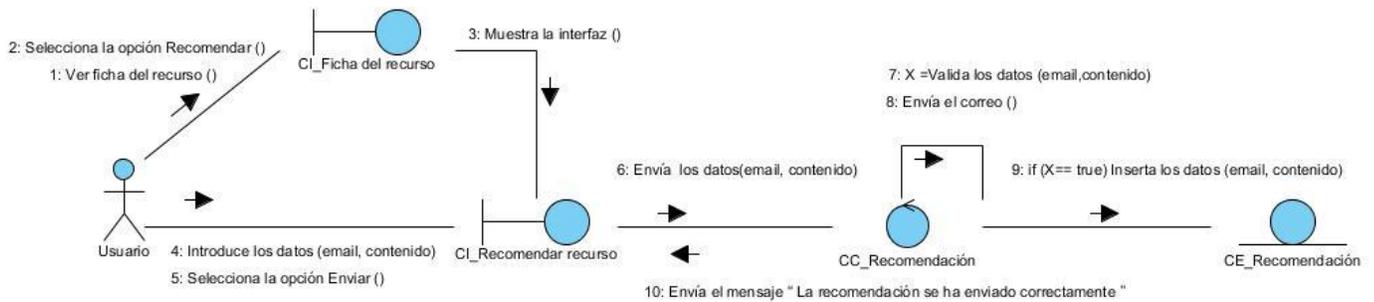


Figura 13. DC del CU: Recomendar recurso educativo

3.2 Modelo del diseño

El modelo del diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en los requisitos funcionales y no funcionales, junto a otras restricciones relacionadas con el entorno de implementación. (20)

3.2.1 Patrones de diseño utilizados

Los patrones de diseño representan soluciones a problemas que surgen cuando se desarrolla un software en un contexto particular. *“Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí adaptada para resolver un problema de diseño general en un contexto particular”* (39). Para realizar el diseño de la aplicación se han utilizado algunos de los patrones existentes, los cuales se mencionan a continuación:

- **Patrones GRASP**

Los Patrones Generales de Software para Asignar Responsabilidades (GRASP, por sus siglas en inglés), son parejas de problema y solución, que codifican buenos principios y sugerencias relacionados frecuentemente con la asignación de responsabilidades. Describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. Los patrones GRASP utilizados en la investigación son:

Experto: permite asignar una responsabilidad al experto en información, a la clase que cuenta con la información necesaria para cumplir la responsabilidad (40). Symfony2 incluye la librería ORM Doctrine como interfaz de comunicación con las clases del modelo, lo que permite encapsular toda la lógica de los datos y generar clases para manipular la información de las entidades de la base de datos. Las clases generadas contienen toda la información necesaria de la entidad que

representan y sus funcionalidades, además estas generalmente se encuentran bajo el nombre *entity_nameRepository*. Cada entidad tiene asociado un repositorio, en el módulo se evidencia, en la entidad *LearningResource* con su respectivo *LearningResourceRepository*.

- **Bajo Acoplamiento:** este patrón consiste en asignar una responsabilidad para mantener el bajo acoplamiento. El acoplamiento es una medida de la fuerza con que una clase está conectada con otras clases, de cómo se conocen y recurren a ellas (40). El propósito de este patrón es que exista entre las clases la menor dependencia posible, para en caso de producirse una modificación en alguna de ellas, no afecte las restantes. Se evidencia mediante las clases que implementan la lógica de acceso a datos que se encuentran en el modelo, las cuales no presentan asociaciones con las clases de las vistas o las controladoras.
- **Alta Cohesión:** la solución que propone este patrón consiste en asignar una responsabilidad de modo que la cohesión siga siendo alta. La cohesión es una medida de cuanto están relacionadas y enfocadas las responsabilidades de una clase (40). Se evidencia mediante la implementación de la clase controladora ***SuggestionController.php***, la cual está formada por diferentes funcionalidades en la que se encuentra el método ***showAction()***, en donde se realiza búsqueda de datos sobre la entidad ***Suggestion*** y el resultado se muestra en la vista ***show.html.twig***.

```

public function showAction($id) {
    $em = $this->getDoctrine()->getManager();
    $suggestion = $em->getRepository('RhodaLearningResourceBundle:Suggestion')->find($id);
    if (!$suggestion) {
        throw $this->createNotFoundException('No hay sugerencias');
    } else {
        $context = $this->get('security.context');
        if ($context->isGranted('IS_AUTHENTICATED_FULLY') == false) {
            throw new AccessDeniedException();
        }
    }
    return $this->render('RhodaLearningResourceBundle:Suggestion:show.html.twig', array(
        'suggestion' => $suggestion
    ));
}

```

Figura 14. Ejemplo de utilización del patrón Alta Cohesión

- **Creador:** la solución del mismo consiste en asignar a una clase B la responsabilidad de crear instancia de una clase cuando se esté en presencia de los siguientes casos (40):

- B agrega los objetos de A.
- B contiene los objetos de A.
- B registra las instancias de los objetos de A.
- B utiliza específicamente los objetos de A.
- B tiene los datos de inicialización que son transmitidos a la clase A cuando este objeto sea creado.

El propósito fundamental de este patrón consiste en guiar la asignación de responsabilidades relacionada con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. Su utilización se evidencia mediante las clases controladoras las cuales crean instancias de las entidades del modelo.

```
public function importAction()
{
    $request = $this->getRequest();
    $em = $this->getDoctrine()->getManager();
    $select = $request->get('type');
    $col = $request->get('collection');
    $learning = new LearningResource();
    $form = $this->createForm(new LearningResourceType(), $learning);
    $submit = $request->get('aceptar');
```

Figura 15. Ejemplo de utilización del patrón Creador

- **Controlador:** propone como solución asignar la responsabilidad de recibir o manejar un mensaje de los eventos del sistema a otra clase que representa una de las siguientes opciones (39):
 - representa el sistema global (controlador de fachada).
 - representa un escenario de caso de uso en el que tiene lugar un evento del sistema.

La solución de este patrón se evidencia mediante las clases controladoras, las cuales cuentan con las funcionalidades encargadas del manejo de los procesos que se realizan en el sistema.

3.2.2 Diagramas de clases del diseño

Los diagramas de clases del diseño (DCD), corresponden a los casos de uso del sistema y muestran el diseño estático a través de las clases, subsistemas y relaciones que participan. En el presente subepígrafe se presentan los DCD correspondientes a los CU: Importar recurso educativo, Descargar recurso educativo y Recomendar recurso educativo, los restantes DCD se encuentran en el **Anexo III**.

Para una mejor comprensión de los diagramas, se describe el DCD asociado al CU: Importar recurso educativo. En el mismo las clases son agrupadas por paquetes basándose en la estructura del patrón Modelo Vista Controlador. El paquete Vista agrupa las interfaces asociadas al usuario, mientras el paquete Controlador contiene una SP (Página Servidora) la cual mediante los métodos implementados accede a las clases del paquete del Modelo para dar respuestas a las peticiones del usuario. El paquete ORM Doctrine es utilizado para la comunicación de las clases del modelo con las entidades de la base de datos.

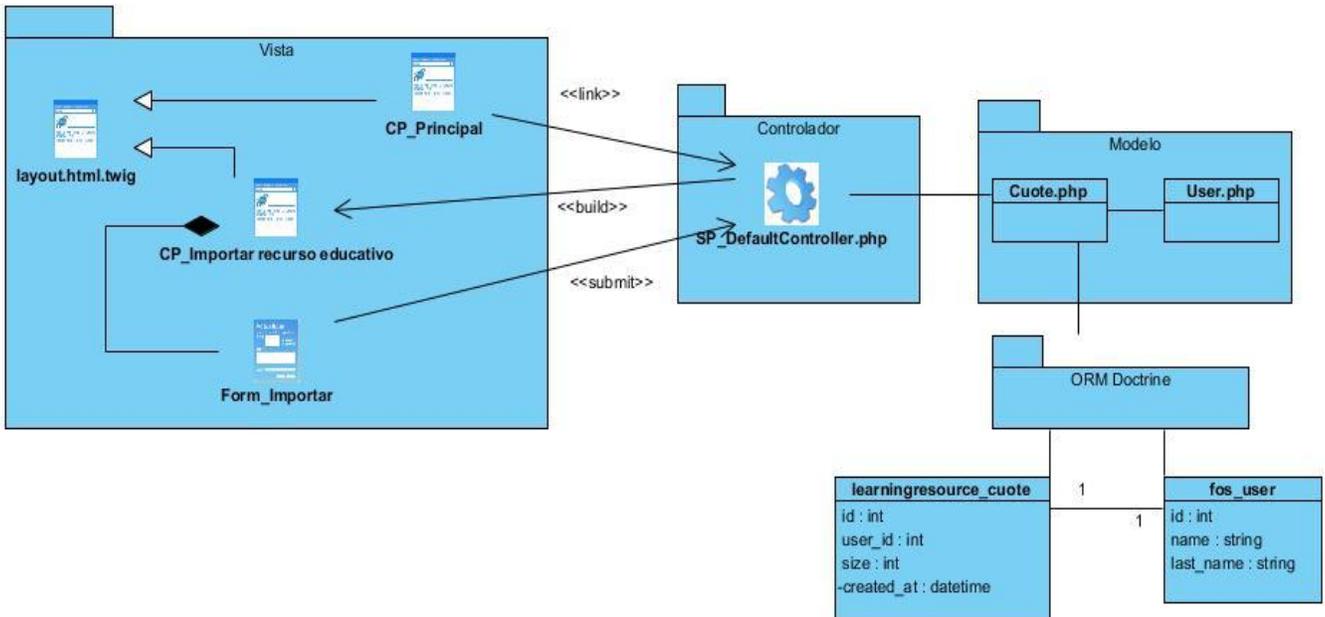


Figura 16. DCD del CU: Importar recurso educativo

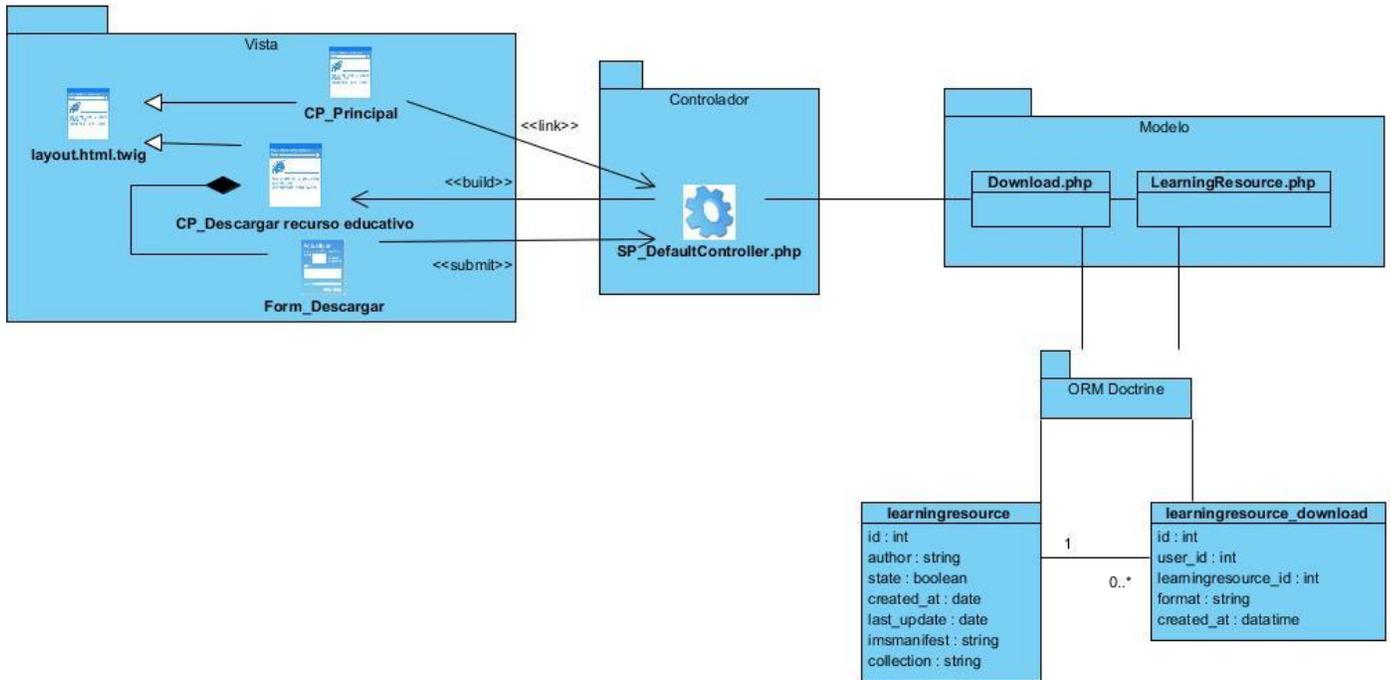


Figura 17. DCD del CU: Descargar recurso educativo

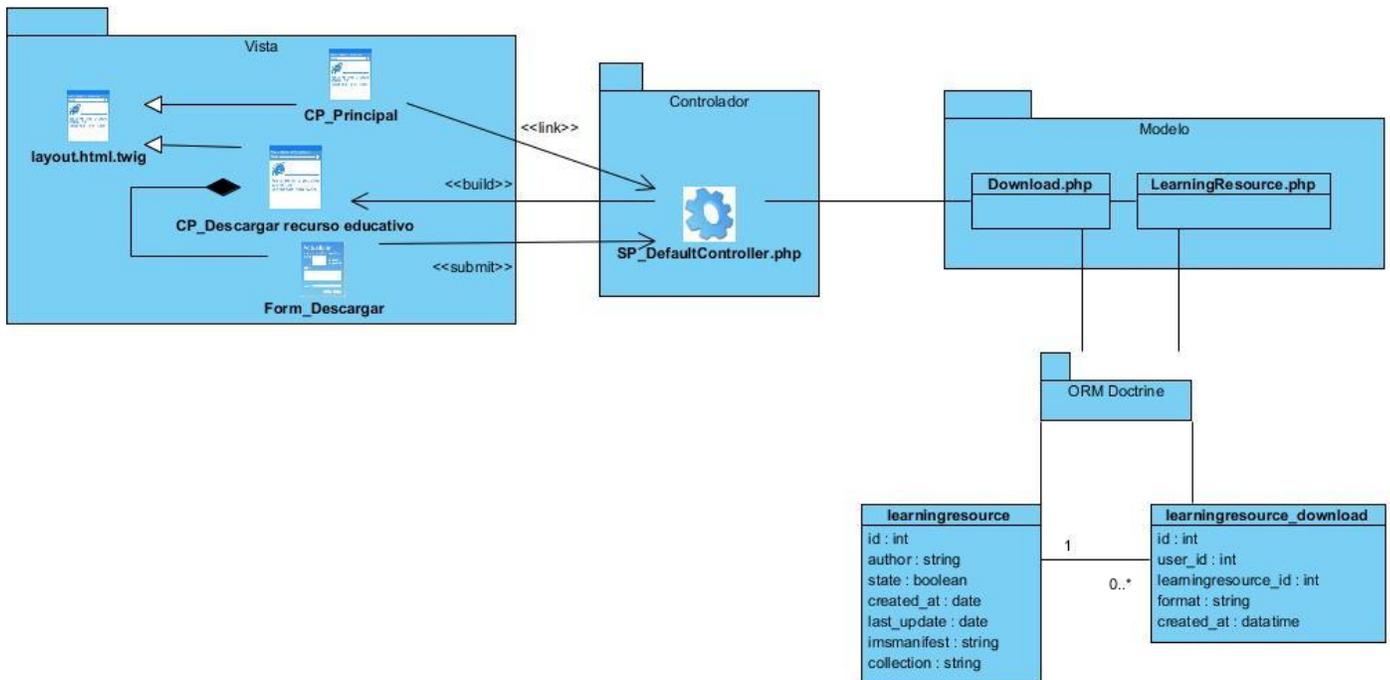


Figura 18. DCD del CU: Recomendar un recurso educativo a otro usuario

3.3 Modelo de datos

El modelo de datos constituye la representación de un fenómeno de la realidad objetiva a través de los objetos, sus propiedades y las relaciones que se establecen entre ellos. El mismo está compuesto por tres piezas fundamentales: el objeto de datos, los atributos y las relaciones entre las que se conectan. (41)

Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, se describe en el siguiente modelo de datos la representación lógica de los datos del sistema.

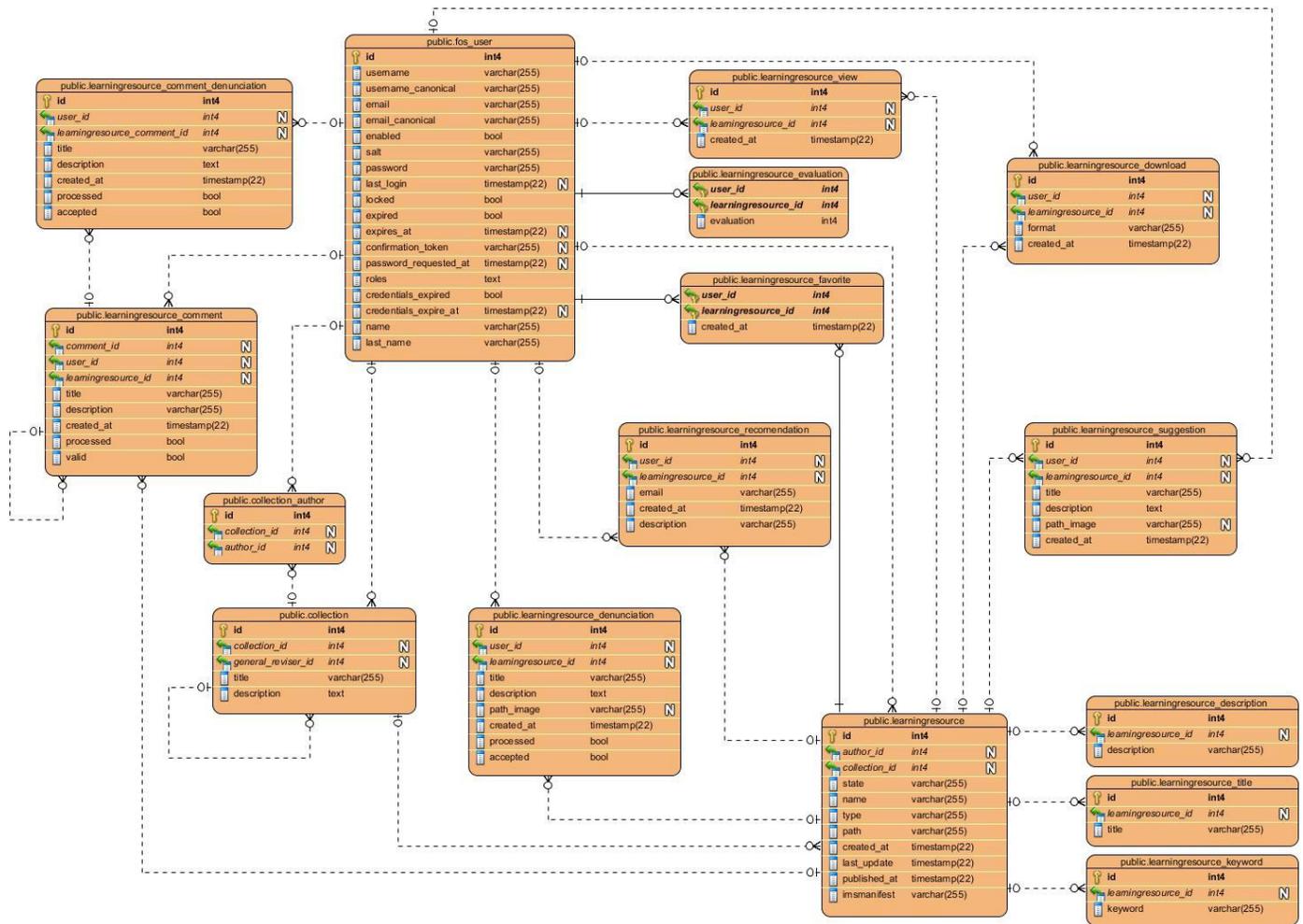


Figura 19. Modelo de datos

Conclusiones parciales

El análisis como flujo de trabajo contribuyó a obtener una vista estática del sistema, lo que facilitó la comprensión en profundidad de los requisitos del módulo. El modelo del diseño obtenido permitió describir la realización física y dinámica de los casos de uso y crear de esta manera, una entrada apropiada para las actividades de implementación.

Capítulo 4. Implementación y pruebas

Introducción

Una vez realizado el flujo de trabajo correspondiente al análisis y diseño, se tiene una visión más general de cómo se implementarán las funcionalidades del sistema. El propósito de este flujo permitirá describir de manera general la implementación de todas las clases y objetos asociados al sistema, teniendo como resultado un módulo ejecutable. Luego de concluida la etapa anterior se decide llevar a cabo las pruebas del sistema con el objetivo de verificar todo el proceso de desarrollo. En la etapa de implementación y pruebas se tienen en cuenta todos los elementos arquitectónicos definidos con anterioridad y permite concluir con éxito el proceso de desarrollo de la propuesta de solución.

4.1 Estándar de codificación

Los estándares de codificación son las reglas que comprenden todo lo relacionado al código fuente de cualquier software, por lo que resulta importante establecer su utilización para asegurar que durante el desarrollo del mismo cualquier programador pueda entender el código y trabajar de forma coordinada.

Para el desarrollo módulo en cuestión se hace referencia a las pautas de codificación definidas por el centro FORTES para el marco de trabajo Xalix. De las mismas se utiliza la siguiente:

- Estilo *camelCase*, para nombrar funciones y métodos. Los nombres de las funciones y métodos son palabras compuestas, por lo que se comienzan a escribir con letra minúscula la primera palabra y a partir de la segunda palabra con letra mayúscula.
- Se utiliza el guión bajo (“_”), como separador de los nombres compuestos de las entidades en la base de datos.

4.2 Modelo de implementación

El modelo de implementación describe cómo los elementos del modelo del diseño, como las clases, se implementan en términos de componentes, ficheros de código fuente y ejecutables. Además, define una jerarquía de subsistemas de implementación que contiene componentes e interfaces. (20)

4.2.1 Diagrama de componentes

Los diagramas de componentes son usados para mostrar los componentes de software y las relaciones lógicas entre ellos en un sistema.

Los estereotipos utilizados para representar dichos componentes son:

- <<file>>: ficheros de configuración o código fuente.
- <<library>>: representan una librería estática o dinámica.

El siguiente diagrama es la representación del módulo de gestión de recursos educativos, dividido según propone el patrón arquitectónico utilizado MVC. El paquete *Views*, agrupa los subpaquetes: *Comment*, *Default*, *Includes* y *Suggestion* que contienen los componentes asociados a las vistas HTML, las cuales importan los componentes de los paquetes CSS y JS. En el paquete *Controller* se encuentran los componentes relacionados a las clases controladoras: *CommentController.php*, *DefaultController.php* y *SuggestionController.php*. Por último, en el paquete *Entity* se encuentran las clases del modelo, utilizando el componente Doctrine para mapear los elementos de la base de datos.

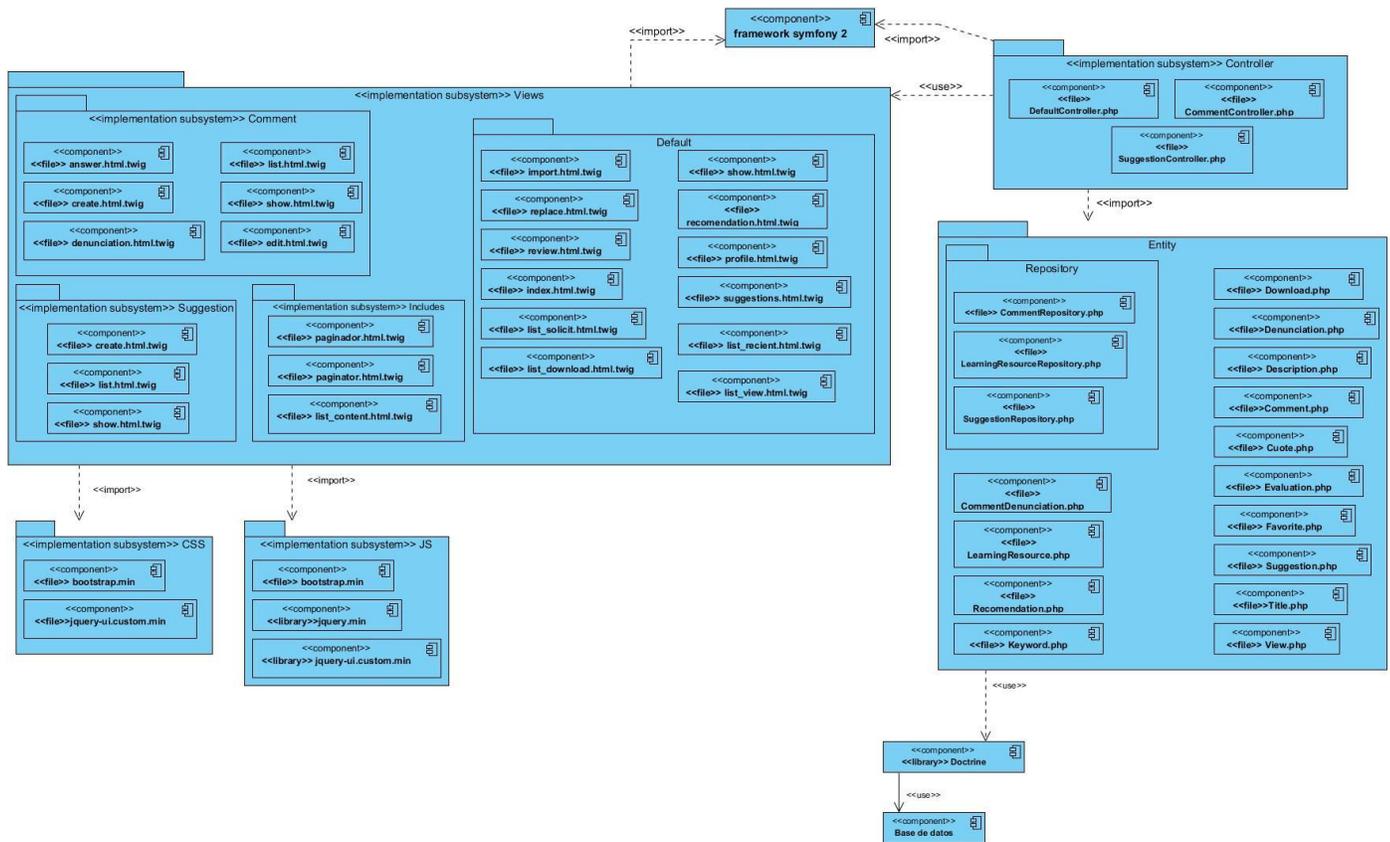


Figura 20. Diagrama de componentes

4.2.2 Modelo de despliegue

El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. En el mismo se puede observar lo siguiente (20):

- Cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar.
- Los nodos poseen relaciones que se representan medios de comunicación entre ellos, tales como: internet, intranet, bus, y similares.
- Representa una correspondencia entre la arquitectura del software y la arquitectura del hardware.

En la siguiente figura se describe el diagrama de despliegue correspondiente al sistema RHODA, debido a que el módulo implementado forma parte de dicho sistema se tiene en cuenta en la presente investigación.

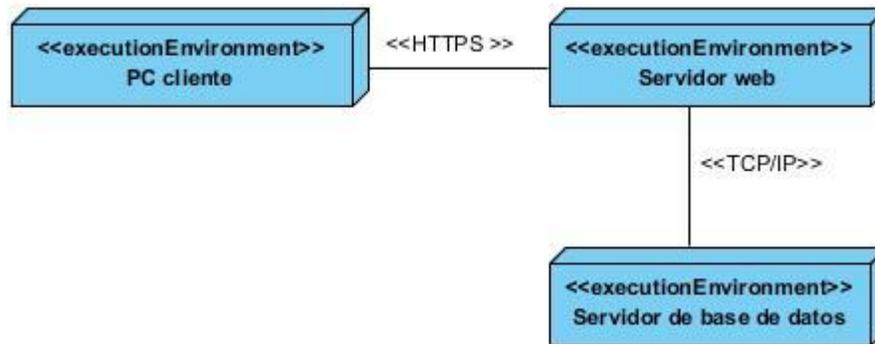


Figura 21. Modelo de despliegue

Descripción del diagrama de despliegue

- PC cliente: representa el nodo de la interacción de los usuarios con la aplicación a través de un navegador web.
- Servidor web: nodo que representa el servidor donde se encuentra la aplicación web.
- Servidor de base de datos: nodo que representa el servidor donde estará la base de datos de la aplicación.

4.3 Pruebas

Con el objetivo de controlar que los productos cumplan con los requisitos funcionales y determinar la calidad de los mismos, resulta importante realizar pruebas. Estas constituyen una actividad en la cual un sistema o unos de sus componentes se ejecutan en circunstancias previamente especificadas, donde los resultados se observan, se registran y se realiza una evaluación de algún aspecto del sistema. Dentro de los procedimientos para la realización de pruebas al software, se encuentran las pruebas de caja blanca y las pruebas de caja negra.

4.3.1 Pruebas de caja negra

Las pruebas de caja negra, también conocidas como pruebas de comportamiento, se concentran en los requisitos funcionales del software. Una prueba de este tipo examina algún aspecto funcional del sistema

que tiene poca relación con la estructura lógica interna del software. Se encargan de encontrar errores de: funciones incorrectas o faltantes, de interfaz, en estructuras de datos o el acceso a base de datos externas, errores de comportamiento y errores de inicialización. (36)

Existen varios métodos para realizar las pruebas de caja negra, entre los que se encuentran, según Pressman (36):

- **Método de partición equivalente:** divide el dominio de entradas de un programa en clases de datos, a partir de las cuales pueden derivarse casos de prueba.
- **Método del análisis de valores límites:** se encarga del análisis de los valores límites en los campo de entrada.
- **Método de prueba de la tabla ortogonal:** se aplica en problemas donde el dominio de entrada es relativamente pequeño, el método resulta útil para encontrar fallas de región (categoría de error asociado con los defectos de la lógica en un componente de software).

Para la verificación del correcto funcionamiento del módulo de gestión de recursos educativos se realizaron pruebas de caja negra, aplicando el método de partición equivalente.

Partiendo de las descripciones de los 28 CU, se generaron los DCP correspondientes, se confeccionó cada caso de prueba con la identificación de los principales escenarios en dependencia de cada acción del actor.

A continuación, se muestra el DCP asociado al CU: Recomendar un recurso educativo a otro usuario. Los restantes se encuentran en el expediente del proyecto.

DCP del CU: Recomendar recurso educativo a otro usuario.

Descripción general

El caso de uso inicia una vez que el usuario accede a la interfaz Recomendar un recurso educativo y selecciona la opción Recomendar. El sistema muestra un formulario donde se puede insertar la dirección de correo a quien se quiere enviar la recomendación y una breve descripción.

Tabla 5. DCP del CU: Recomendar recurso educativo a otro usuario

Escenario	Descripción	Correo electrónico	Contenido	Respuesta del sistema	Flujo central
EC 1.1 Seleccionar la opción Recomendar	Permite recomendar un recurso educativo a otro usuario.	N/A	N/A	Muestra la interfaz que permite recomendar un recurso educativo, la misma contiene los siguientes campos: <ul style="list-style-type: none"> • Correo electrónico: campo de texto que permite insertar una dirección de correo electrónico. • Contenido: campo de texto que permite insertar un mensaje. Además muestra las siguientes opciones: Enviar: botón que permite enviar el correo al destinatario. Cancelar: botón que permite retornar a la ficha	Inicio/ Mostrar recurso educativo / Recomendar recurso educativo.

Capítulo 4. Implementación y Pruebas
Módulo para la gestión de recursos educativos en RHODA

				del recurso educativo.	
EC 1.2 Introduce los datos necesarios y selecciona la opción Enviar.	Insertar los datos correspondientes a los campos: correo electrónico y contenido.	V	V	<ul style="list-style-type: none"> • Valida los campos. • Envía la recomendación al usuario destinatario. • Muestra el mensaje “La recomendación se ha enviado correctamente”. 	Inicio / Mostrar recurso educativo / Recomendar recurso educativo.
EC 1.3 Existen campos vacíos.	No se inserta ningún dato en los campos obligatorios.	I	V	<ul style="list-style-type: none"> • Muestra el mensaje “Campos vacíos”. • Permite introducir nuevamente los datos. 	Inicio / Mostrar recurso educativo / Recomendar recurso educativo.
		V	I		
		I	I		
EC 1.4 Formato de correo electrónico incorrecto.	La dirección de correo electrónica introducida no es correcta.	I	V	Muestra el mensaje “Introduzca una dirección de correo válida ”	Inicio / Mostrar recurso educativo / Recomendar recurso educativo.
EC 1.5	Permite cancelar la	V	V	Retorna a la interfaz	Inicio /

Opción Cancelar.	acción de recomendar un recurso educativo a otro usuario.	I	V	donde la ficha del recurso educativo.	Mostrar recurso educativo / Recomendar recurso educativo.
		V	I		

Nota:

Las celdas de la tabla contienen V, I, o N/A, lo que indica válido, inválido y no aplica, respectivamente.

En la siguiente tabla se muestra la descripción de las variables que intervienen en el DCP, así como los valores válidos que las mismas pueden tomar.

Descripción de las variables

Tabla 6. Descripción de las variables del CU: Recomendar recurso educativo a otro usuario.

No	Nombre del campo	Clasificación	Valor nulo	Descripción
1.	Correo electrónico	campo de texto	No	Formato: (usuario@dominio) Ejemplo: <ul style="list-style-type: none"> preyes@estudianates.uci.cu preyes@uci.cu
2.	Contenido	campo de texto	No	Formato (alfabético). Ejemplo: Te recomiendo.

4.4 Resultados de las pruebas

Como resultado de las pruebas realizadas una vez concluida la implementación del módulo, se detectaron un conjunto de no conformidades, las cuales fueron clasificadas en: significativas, no significativas y recomendaciones. En la siguiente tabla se muestran los resultados obtenidos luego de aplicadas las pruebas al módulo desarrollado:

Tabla 7. Cantidad de no conformidades por iteración

Clasificación de las No Conformidades	1era Iteración	2da Iteración	3ra iteración
Significativas	10	5	0
No significativas	6	2	0
Recomendaciones	4	3	2

Para una mejor comprensión de los resultados, se muestra en el siguiente gráfico la cantidad de no conformidades encontradas en cada una de las iteraciones realizadas.

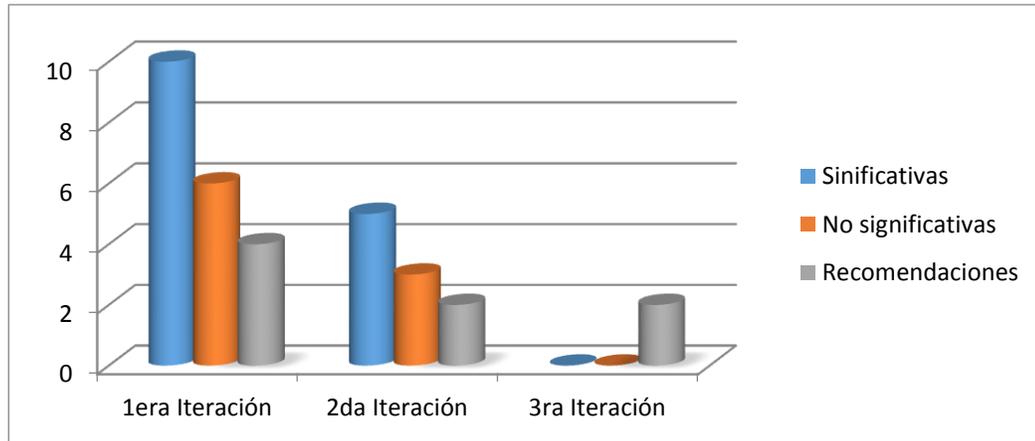


Figura 22. Resultados de las iteraciones

Las no conformidades detectadas en cada iteración fueron analizadas y resultas por el equipo de desarrollo. Las pruebas se realizaron de forma iterativa e incremental, logrando que para cada iteración fueran corregidos los errores detectados en las iteraciones anteriores, lo que contribuyó a mejorar la calidad y funcionalidad del módulo implementado.

Conclusiones parciales

El modelo de implementación obtenido permitió describir cómo los elementos del modelo de diseño se implementan en términos de componentes. El método de prueba seleccionado posibilitó detectar los errores existentes al concluir la implementación del módulo, los cuales fueron revisados y corregidos para garantizar la calidad del mismo.

Conclusiones generales

La investigación realizada permite arribar a las siguientes conclusiones:

- ✓ El estudio de las tendencias relacionadas a la gestión de recursos educativos en repositorios digitales, permitió identificar las funcionalidades básicas a incorporar en el módulo para la gestión de recursos educativos en RHODA.
- ✓ Mediante el análisis y diseño se generaron los artefactos necesarios, que fueron utilizados como base para las actividades de implementación.
- ✓ El módulo desarrollado permite a los profesores de la universidad, la gestión, visualización y almacenamiento de sus recursos, contribuyendo además, a elevar su calidad, reutilización, durabilidad y accesibilidad, mediante el uso de los estándares SCORM, IMS CP y LOM.
- ✓ Las pruebas realizadas permitieron identificar y erradicar las deficiencias existentes en el funcionamiento del módulo, evitando así, insatisfacciones por parte de los usuarios.

Recomendaciones

A partir del trabajo realizado se recomienda a los miembros del equipo de desarrollo de RHODA:

- ❖ Incorporar al módulo desarrollado los estándares de catalogación: DublinCore, CanCore e IMS-MD.

Referencias bibliográficas

1. **RedTTnet.** *La Formación Sin Distancia.* España : s.n., 2012.
2. **González, Ing. Roxana Cañizarez.** *Repositorio de Recursos Educativos para las Instituciones de Educación Superior.* La Habana : s.n., 2012.
3. Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura. [En línea] 2002. [Citado el: 23 de mayo de 2013.] <http://www.unesco.org/new/es/communication-and-information/access-to-knowledge/open-educational-resources/>.
4. **Atkins, D,E. Seel Y Brown, J. and L A Hammond.** *A Review of the Open Educational Resources (OER) Movement: Achievements, Challenges, and New Opportunities.* 2007.
5. **Maurizi, María Rosa.** Que son los Recursos Educativos Abiertos. [En línea] julio de 2012. [Citado el: 10 de mayo de 2014.] <http://www.slideshare.net/mariarosamaurizi/que-son-los-recursos-educativos-abiertos>.
6. **Zapata, Marta.** Programa Integración de Tecnologías a la Docencia. [En línea] septiembre de 2012. [Citado el: 23 de mayo de 2014.] <http://aprendeonline.udea.edu.co/boa/contenidos.php/d211b52ee1441a30b59ae008e2d31386/845/estilo/aHR0cDovL2FwcmVuZGVlbnVhLnVhZWR1LmNvL2VzdGlsc3MvYXp1bF9jb3Jwb3JhdGI2by5jc3M=/1/contenido/>.
7. **Fuente, Gema Bueno de la.** *Modelo de repositorio institucional de contenido educativo (RICE): la gestión de materiales digitales de docencia y aprendizaje en la biblioteca universitaria.* España : s.n., 2010 .
8. **Guzmán, Clara López.** *Los Repositorios de Objetos de Aprendizaje como soporte para entornos e-learning.* Salamanca: España : Tesis de grado, 2005.
9. **Cañizares, R., Febles, J.P.and Estrada, V.** *Los objetos de aprendizaje, una tecnología necesaria para las instituciones de la educación superior en Cuba.* La Habana : s.n., 2012.
10. **Española, Real Academia.** Diccionario de la lengua española. [En línea] [Citado el: 1 de febrero de 2014.] <http://lema.rae.es/drae/?val=repositorio>.
11. **Volder, Carolina de.** SciELO- Scientific Eletronic Library Online. [En línea] Los repositorios de acceso abierto en Argentina: situación actual., 2014. [Citado el: mayo de 10 de 2014.] http://www.scielo.org.ar/scielo.php?script=sci_arttext&pid=S1851-17402008000200005. 1851-1740.

12. **Nacional, Ministerio de educación.** *Recursos Educativos Digitales Abiertos* .
[<http://www.colombiaaprende.edu.co/reda/REDA2012.pdf>] Cundinamarca, Colombia : Graficando Servicios Integrados, 2012. ISBN: 978-958-691-476-5.
13. **The Gateway is a jes & co.** Gateway to 21st Century Skills. [En línea] [Citado el: 12 de Febrero de 2014.] <http://www.thegateway.org/>.
14. **Hodgins, W., Latson, C., Brown, J., Ramírez, G.T., Torres, A., Christensen, M., Damme, L.V., Meyers, D., Riddle, S., Metcalf, D., and Huffman, K., Lozeau, D. AND Cameron, R.** Making Sense of Learning Specifications & Standards: A Decision Maker's Guide to their Adoption. [En línea] 2002. www.masie.com.
15. **Manjón, Baltasar Fernández.** CNICE Red Digital. [En línea] 2006. [Citado el: 10 de Abril de 2014.] http://reddigital.cnice.mec.es/6/Articulos/pdf/Articulos_2.pdf. 1696-0823.
16. —. [En línea] 2006. [Citado el: 31 de Enero de 2014.] http://reddigital.cnice.mec.es/6/Articulos/articulo_capitulo.php?articulo=2&capitulo=1.
17. **Manjón, Baltasar Fernández, Ger, Pablo Moreno, Rodríguez, José Luis Sierra and Ortíz, Martínez.** *Uso de estándares aplicados a Tic en educación*.
18. **Lozano, María Pilar Laguna.** Resumen del Modelo de Referencia SCORM 2004 (4ta Edition). [En línea] [Citado el: 10 de Abril de 2014.] http://unizar.es/innovacion/ecoleccion1/archivos/PDF/55_SCORM_RESUMEN.pdf.
19. **ADL.** Advanced Distributed Learning Emerging and Enabling Technologies for the Design of Learning Object Repositories. [En línea] 2002. [Citado el: 22 de Marzo de 2014.] <http://xml.coverpages.org/ADLRepositoryTIR.pdf>.
20. **JACOBSON, Ivar, BOOCH, Grady and RUMBAUGH, James.** *El Proceso Unificado de Desarrollo de Software*. Madrid : s.n., 2000. ISBN 84-7829-036-2..
21. **Álvarez, Jorge Cortés.** [En línea] [Citado el: Junio de 9 de 2014.] <http://es.slideshare.net/cortosalvarez/metodologa-rup>.
22. **López Ramiro Pecho, Ballesterero Julio César.** Herramientas CASE. [En línea] 2008. [Citado el: 27 de Febrero de 2014.] <http://tpsis324.blogspot.com/>.
23. **Manager, Free Download.** Free Download Manager. [En línea] 2008. [Citado el: 22 de Mayo de 2014.] http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p/.

24. **GRUPO ENTEL. Cientec.** . CIENTEC Grupo ENTEL . [En línea] [Citado el: 27 de Febrero de 2014.] <http://www.cientec.com/analisis/ana-uml.html>.
25. **The PHP Group.** <http://www.php.net/manual/es/intro-whatism.php>. [En línea]
26. **PHP.** ¿Qué es PHP? - Manual. [En línea] [Citado el: 8 de Febrero de 2014.] <http://www.php.net/manual/es/intro-whatism.php>.
27. **Tuero, Laura.** Go-moviles.es. [En línea] Marzo de 2013. [Citado el: 9 de Febrero de 2014.]
28. **Eguiluz, Javier.** Introducción a JavaScript. [En línea] [Citado el: 9 de Febrero de 2014.] <http://librosweb.es/javascript/index.html>.
29. **PC-SERVEIS.** . [En línea] 2010. [Citado el: 9 de Febrero de 2014.] <http://www.blog.pc-serveis.es/desarrollo-web/manual-de-hojas-css3-introduccion>.
30. **Labs, Sensio.** *The Book*. 2013.
31. **Álvarez, Miguel Angel.** Manual de jQuery. [En línea] 2009. <http://www.desarrolloweb.com/manuales/manual-jquery.html>.
32. **Borillo, Ricardo.** GENBETA: dev desarrollo y software . [En línea] 23 de Febrero de 2012. [Citado el: 9 de Febrero de 2014.] <http://www.genbetadev.com/desarrollo-web/disenando-tu-nuevo-proyecto-web-con-boot>.
33. **Alvarez, Sara.** Sistemas gestores de bases de datos. [En línea] 31 de Julio de 2007. [Citado el: 9 de Febrero de 2014.] <http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
34. **PostgreSQL: Dossier de Prensa de PostgreSQL 9.0.** [En línea] <http://www.postgresql.org/about/press/presskit90/es/>.
35. Una Introducción a Apache . [En línea] [Citado el: 12 de Febrero de 2014.] http://linux.ciberaula.com/articulo/linux_apache_intro.
36. **Pressman, Roger S.** *Ingeniería de Software. Un enfoque práctico* . 2005. . ISBN 0072853182.
37. **Zaninotto, F, Potencier, F.** Symfony 1.1, la guía definitiva. [En línea] 2009. [Citado el: 7 de Abril de 2014.] http://librosweb.es/symfony_1_1.
38. **Övergaard, Gunnar and Palmkvist, Karin.** *Use Cases Patterns and Blueprints*. s.l. : Addison Wesley Professional, 2004. ISBN 0-13-145134-0.
39. **E. Gamma, R. Helm, R., Johnson, and J., Vlissides and E. GAMMA, R.** Design Patterns . [En línea] 1995. <http://siul02.si.ehu.es/~alfredo/iso/06Patrones.pdf>.
40. **Larman, Craig.** *UML y Patrones. Introducción al análisis y diseño orientado a objetos* . Mexico : s.n., 1999. 970-17-0261-1. .

41. **García, Lic. Rosa María Mato.** *Diseño de bases de datos* . 1999.
42. **Ávarez, Miguel Angel.** Manual de jQuery. [En línea] 2009 .
<http://www.desarrolloweb.com/manuales/manual-jquery.html>.

Anexo I. Descripción de los casos de uso

DCU_1: Listar recursos educativos por categoría.

Objetivo	Listar los recursos educativos que han sido almacenados en el repositorio.	
Actores	Invitado (inicia).	
Resumen	El caso de uso inicia una vez que el invitado accede a la vista principal del sistema. Podrá listar los recursos almacenados en el repositorio por diferentes criterios.	
Complejidad	Baja	
Prioridad	Secundario	
Precondiciones	Que existan al menos cinco recursos educativos en el sistema.	
Postcondiciones	Se han listado los recursos educativos almacenados en el sistema.	
Flujo de eventos		
Flujo básico: Listar recursos educativos por categoría		
	Actor	Sistema
1	Accede al sistema.	1.1 Muestra la vista principal del sistema, encontrándose en el final de la misma los últimos cinco recursos agrupados por las categoría: <ul style="list-style-type: none"> • <i>Recursos más recientes.</i> • <i>Recursos más visualizados.</i> • <i>Recursos más solicitados.</i> • <i>Recursos más descargados.</i> Además permite listar todos los recursos, accediendo al enlace “Ver más”, que contiene cada categoría.
2	Accede a la opción “Ver más”	2.1 Muestra el listado de los recursos educativos y los detalles asociados al mismo.
		3.1 Termina el caso de uso.
Requisitos	RF1, RF2, RF3, RF4.	

funcionales	
-------------	--

CU_3: Exportar recurso educativo.

Objetivo	Exportar un recurso educativo en los formatos: SCORM 1.2, 2004, recurso y el manifiesto asociado al recurso	
Actores	Invitado (inicia).	
Resumen	El caso de uso se inicia cuando el invitado selecciona la opción de exportar un recurso educativo y finaliza cuando es exportado.	
Complejidad	Alta.	
Prioridad	Crítico.	
Precondiciones	Debe haberse ejecutado el CU: Ver ficha de un recurso educativo.	
Postcondiciones	El recurso educativo es exportado.	
Flujo de eventos		
Flujo básico : Exportar recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Exportar</i>	1.1 Muestra las opciones a las que se puede exportar el recurso: <ul style="list-style-type: none"> • SCORM 1.2 • SCORM 2004 • Manifiesto • Recurso
2.	Selecciona el formato en que desea exportarlo.	2.1 Exporta el recurso educativo en la dirección seleccionada en formato .ZIP
3.		3.1 Termina el caso de uso.
Requisitos funcionales	RF13	

CU_5: Ver comentarios de un recurso educativo

Objetivo	Ver comentarios de un recurso educativo.	
Actores	Invitado (inicia)	
Resumen	El caso de uso inicia una vez que el invitado selecciona la opción <i>Comentar</i> en la ficha de un recurso educativo y finaliza cuando visualizan los comentarios.	
Complejidad	Baja.	
Prioridad	Alta.	
Precondiciones	Se haya ejecutado el CU: Ver ficha de un recurso educativo.	
Postcondiciones	Se visualizan los comentarios.	
Flujo de eventos		
Flujo básico: Ver comentarios de un recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Comentar</i> .	<p>1.1 Muestra una interfaz con los siguientes datos:</p> <ul style="list-style-type: none"> • Título: muestra el título del comentario. • Descripción: muestra el contenido del comentario. <p>Además permite realizar las siguientes opciones:</p> <p>Denunciar: permite denunciar un comentario de un recurso educativo. Ver CU: <i>Denunciar comentario de un recurso educativo</i>.</p> <p>Editar: permite editar los datos de un comentario. Ver CU: <i>Editar comentario de un recurso educativo</i>.</p> <p>Responder: permite responder un comentario. Ver CU: <i>Responder comentario de un recurso educativo</i>.</p> <p>Adicionar: botón que permite adicionar un comentario a un recurso educativo. Ver CU: <i>Adicionar comentario a un recurso educativo</i>.</p>
2.		2.1 Termina el caso de uso.

Requisitos funcionales	RF6
-------------------------------	-----

CU_6: Ver ficha de un recurso educativo

Objetivo	Ver ficha de un recurso educativo.	
Actores	Invitado (inicia)	
Resumen	El caso de uso inicia una vez que el invitado selecciona un recurso educativo y finaliza cuando visualiza la ficha del recurso.	
Complejidad	Baja.	
Prioridad	Alta.	
Precondiciones	Que exista al menos un recurso educativo en el sistema.	
Postcondiciones		
Flujo de eventos		
Flujo básico: Ver ficha del recurso educativo		
	Actor	Sistema
1.	Selecciona el recurso educativo.	<p>1.2 Muestra la ficha del recurso educativo. La misma contiene los siguiente información:</p> <ul style="list-style-type: none"> • Autor: nombre del autor del recurso educativo. • Colección: nombre de la colección a la que pertenece el recurso educativo. • Calificación: muestra la calificación promedio del recurso educativo. • Fecha de publicación: muestra la fecha de publicación del recurso. • Sugerencias: muestra las sugerencias que tiene el recurso educativo. • Palabras claves: contiene las palabras claves del recurso educativo. • Descripción: muestra una breve descripción del recurso educativo. <p>Del autor se muestra:</p>

	<ul style="list-style-type: none">• Correo: dirección de correo electrónico.• Publicaciones: muestra la cantidad recursos educativos importados al repositorio. <p>Contiene además las siguientes opciones de acuerdo a los privilegios del sistema.</p> <p><i>Invitado :</i></p> <ul style="list-style-type: none">• Visualizar: botón que permite visualizar el recurso educativo.• Descargar:• Exportar• Comentario. (Ver comentarios solamente) <p>A partir del <i>Usuario</i> y el <i>Administrador</i> muestra las siguientes opciones:</p> <ul style="list-style-type: none">• Comentar: botón que permite realizar un comentario al recurso educativo.• Sugerir: botón que permite realizar una sugerencia al autor de un recurso educativo.• Recomendar: botón que permite enviar una recomendación de un recurso educativo.• Importar: botón que permite importar un recurso educativo.• Calificar: rating de estrellas que permite darle una calificación al recurso educativo.• Compartir con facebook: ícono que permite que contiene un enlace con la red social facebook.• Compartir con twitter: ícono que permite que contiene un enlace con la red social
--	---

		<p>twitter.</p> <ul style="list-style-type: none"> • Adicionar a favorito: ícono que permite adicionar un recurso educativo a favoritos.
2.		2.1 Termina el caso de uso.
Requisitos funcionales	RF5	

CU_7: Visualizar recurso educativo.

Objetivo	Visualizar un recurso educativo.	
Actores	Invitado (inicia).	
Resumen	El caso de uso se inicia cuando el invitado selecciona la opción <i>Visualizar</i> y finaliza cuando se visualiza el recurso educativo.	
Complejidad	Media.	
Prioridad	Crítico.	
Precondiciones	Se debe haber ejecutado el CU: Ver ficha de un recurso educativo.	
Postcondiciones	El recurso educativo es visualizado.	
Flujo de eventos		
Flujo básico: Visualizar recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Visualizar</i> .	<p>1.1 Muestra una vista según el tipo de recurso educativo:</p> <ul style="list-style-type: none"> • Imagen • Video • Documento • Objeto de aprendizaje. Ver sección 1. Visualizar Objeto de aprendizaje.
2.		2.1 Termina el caso de uso.
Flujos alternos		
1.a Visualizar Objeto de aprendizaje		
	Actor	Sistema
1.a		1.1a Muestra la interfaz donde se visualizará el recurso

		<p>educativo, la misma contiene:</p> <p>Árbol de estructura:</p> <ul style="list-style-type: none"> - Cuadro de selección que permite ver la parte del OA que desea ver. - Botón para desplegar y recoger el “Árbol de estructura”. <p>Visualizador del elemento: área donde se muestra el elemento del OA que haya sido seleccionado en el árbol.</p> <ul style="list-style-type: none"> - Atrás: botón de texto que permite ir al elemento anterior. - Siguiente: botón de texto que permite ir al ítem posterior.
		1.2a Termina el caso de uso.
Requisitos funcionales	RF7	

CU_8: Eliminar recurso educativo.

Objetivo	Eliminar un recurso educativo que se encuentre en el repositorio.	
Actores	Administrador (inicia).	
Resumen	El caso de uso inicia cuando el administrador muestra el listado de los recursos educativos a eliminar y finaliza cuando es eliminado.	
Complejidad	Alta.	
Prioridad	Crítico.	
Precondiciones	Debe existir al menos un recurso educativo en el repositorio.	
Postcondiciones	Queda eliminado el recurso educativo.	
Flujo de eventos		
Flujo básico: Eliminar recurso educativo		
	Actor	Sistema
1.	Lista los recursos educativos.	<p>1.1 Muestra una interfaz con el listado de los recursos educativos. Contiene además la opción:</p> <ul style="list-style-type: none"> - Eliminar: botón que permite eliminar el recurso

		educativo.
2.	Selecciona el recurso educativo y la opción <i>Eliminar</i> .	2.1 Muestra el mensaje de confirmación: ¿Está seguro que desea eliminar el elemento seleccionado? Además de las opciones: Aceptar: botón que permite confirmar el proceso de eliminar un recurso. Cancelar: botón que retorna al listado de los recursos educativos.
3.	Selecciona la opción <i>Aceptar</i> .	3.1 Elimina el recurso educativo. 3.2 Muestra el mensaje: “El elemento se ha eliminado correctamente”
4.		4.1 Termina el caso de uso.
Flujos alternos		
2.1a “Cancelar”		
	Actor	Sistema
2.a	Selecciona opción <i>Cancelar</i> .	2.1a Retorna a la vista donde se muestra el listado de los recursos educativos. 2.2a Termina el caso de uso.
Requisitos funcionales	RF11	

CU_9: Editar las propiedades de un recurso educativo.

Objetivo	Editar las propiedades de un recurso educativo.
Actores	Autor (inicia)
Resumen	El caso de uso inicia cuando al autor de un recurso desea editar las propiedades de un recurso de su autoría y finaliza una vez que son editadas sus propiedades.
Complejidad	Media.
Prioridad	Secundaria.
Precondiciones	Que haya ejecutado el CU: Listar publicaciones y que exista un recurso educativo de su autoría en el sistema.
Postcondiciones	Se modifican las propiedades del recurso educativo.

Flujo de eventos		
Flujo básico Editar las propiedades de un recurso educativo		
	Actor	Sistema
1	Selecciona la opción <i>Editar</i> .	<p>1.1 Muestra la interfaz que permite editar las propiedades del recurso:</p> <ul style="list-style-type: none"> • Título: campo de texto que permite modificar el título del recurso educativo. • Descripción: campo de texto que permite modificar la descripción del recurso educativo. • Palabras claves: campo de texto que permite modificar las palabras claves del recurso educativo. <p>Contiene además las siguientes opciones:</p> <ul style="list-style-type: none"> • Aceptar: botón que permite guardar los cambios realizados. • Cancelar: botón que permite retornar a la lista de recursos educativos publicados.
2.	Modifica los campos y selecciona la opción <i>Aceptar</i> .	<p>2.1 Valida los campos.</p> <p>2.2 Muestra el mensaje: “El recurso educativo se ha editado correctamente”.</p>
3.		3.1 Termina el caso de uso.
Flujos alternos.		
2.1a “Existen campos vacíos”		
	Actor	Sistema
2.a		<p>2.1a Muestra el mensaje: “Completa los campos”</p> <p>2.2a Retorna al punto 2.1 del flujo básico.</p>
2.1b “Formato incorrecto”		
2.b		<p>2.1b Muestra el mensaje: “ Formato no válido”</p> <p>2.2b Retorna al punto 2.1 del flujo básico.</p>
1a “Cancelar”		
	Actor	Sistema

1.a	Selecciona la opción <i>Cancelar</i>	1.1a Retorna a la vista mis publicaciones. (Ver CU: <i>Listar publicaciones</i>). 1.2a Termina el caso de uso.
Requisitos funcionales	RF10	

CU_10: Reemplazar recurso educativo.

Objetivo	Reemplazar un recurso educativo que se encuentre en el repositorio.	
Actores	Autor (inicia).	
Resumen	El caso de uso inicia cuando el autor selecciona la opción <i>Reemplazar</i> que se encuentra en el menú de Recurso Educativo y finaliza una vez que se reemplaza el recurso.	
Complejidad	Alta.	
Prioridad	Crítico.	
Precondiciones	Que exista en el sistema un recurso de su autoría.	
Postcondiciones	El recurso es reemplazado.	
Flujo de eventos		
Flujo básico: Reemplazar recurso educativo		
	Actor	Sistema
1	Selecciona la opción <i>Reemplazar</i> del menú Recursos Educativos.	1.1 Muestra la lista de los recursos asociados al autor y detalles de los mismos permitiendo las siguientes opciones: <ul style="list-style-type: none"> - Localizar: campo de tipo file que permite localizar el recurso a reemplazar. - Reemplazar: botón que permite sustituir el recurso educativo.
2	Localiza el recurso y selecciona la opción <i>Reemplazar</i>	2.1 Valida los campos. 2.2 Reemplaza el recurso educativo. 2.3 Muestra el mensaje: "El recurso educativo se ha reemplazado correctamente".
3.		3.1 Termina el caso de uso.
Flujos alternos		

2.1a “Existen campos vacíos”		
	Actor	Sistema
2.a		2.1a Muestra el mensaje “Debe seleccionar un archivo” 2.2a Termina el caso de uso.
Requisitos funcionales	RF12.	

CU_11: Adicionar comentario a un recurso educativo.

Objetivo	Adicionar un comentario a un recurso educativo.	
Actores	Usuario (inicia).	
Resumen	El caso de uso inicia una vez que el usuario accede a la opción: <i>Comentar</i> en la ficha de un recurso educativo. El sistema permite adicionar un comentario. El caso de uso finaliza una vez que el usuario ha adicionado algún comentario.	
Complejidad	Medio	
Prioridad	Crítico	
Precondiciones	El usuario debe estar autenticado en el sistema y debe haberse ejecutado el CU: Ver comentarios de un recurso educativo.	
Postcondiciones	Se adiciona un comentario.	
Flujo de eventos		
Flujo básico: Adicionar comentario a un recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Adicionar</i> .	1.1 Muestra una interfaz con los datos: <ul style="list-style-type: none"> • Título: campo de texto que permite insertar el título del comentario al recurso educativo. • Descripción: campo de texto que permite insertar el contenido del comentario a crear. Aceptar: botón que permite realizar la acción de adicionar.

		<p>Cancelar: botón que permite retornar a la lista de comentarios del recurso.</p> <p>Además muestra las opciones:</p> <ul style="list-style-type: none"> • Desde: campo de texto que permite insertar una fecha de inicio. • Hasta: campo de texto que permite insertar una fecha de fin. <p>Buscar: botón que te permite filtrar los comentarios que tiene un recurso por un rango de fecha.</p>
2.	Selecciona la opción <i>Aceptar</i> .	<p>2.1 Valida que los campos.</p> <p>2.2 Adiciona el comentario.</p> <p>2.3 Muestra el mensaje “Su comentario ha adicionado correctamente”.</p>
3.		3.1 Termina el caso de uso.
Flujos alternos		
2.1a “Existen campos vacíos”		
	Actor	Sistema
2.a		<p>2.1a Muestra el mensaje “Completa este campo”.</p> <p>2.2a Retorna al punto 2.1 del flujo básico.</p>
2.1b “Formato incorrecto”		
	Actor	Sistema
2.b		<p>2.1b Muestra el mensaje “Formato no válido”.</p> <p>2.2b Retorna al punto 2.1 del flujo básico.</p>
1.1a “Cancelar”		
	Actor	Sistema
1.a	Selecciona la opción <i>Cancelar</i> .	1.1a Retorna a la interfaz donde se

		muestran listar los comentarios de un recurso educativo. (Ver CU: <i>Ver comentarios de un recurso educativo</i>) 1.2a Termina el caso de uso.
Requisitos funcionales	RF14	

CU_12: Editar comentario de un recurso educativo.

Objetivo	Editar un comentario realizado a un recurso educativo.	
Actores	Usuario (inicia).	
Resumen	El caso de uso inicia una vez que el usuario accede a la opción: <i>Comentar</i> en la ficha de un recurso educativo. El sistema permite editar un comentario. El caso de uso finaliza una vez que el usuario ha modificado su comentario.	
Complejidad	Medio.	
Prioridad	Crítico.	
Precondiciones	El usuario debe estar autenticado en el sistema y haberse ejecutado el CU: Ver ficha de un recurso educativo.	
Postcondiciones	Se realizó la acción de editar un comentario.	
Flujo de eventos		
Flujo básico Editar comentario de un recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Comentar</i> .	<p>1.1 Muestra:</p> <ul style="list-style-type: none"> • Título el título del recurso educativo. • Descripción del comentario. <p>Además permite realizar las siguientes opciones:</p> <ul style="list-style-type: none"> • Responder. Ver CU: “Responder comentario”. • Denunciar. Ver CU: “Denunciar comentario de un recurso educativo”. • Editar: enlace que permite editar un

		comentario de un recurso educativo.
2.	Selecciona la opción <i>Editar</i> .	<p>2.1 Muestra los campos</p> <ul style="list-style-type: none"> • Título: campo de texto que permite editar el título del recurso educativo. • Descripción: campo de texto que permite editar la descripción del recurso educativo. <p>Además muestra las siguientes opciones:</p> <ul style="list-style-type: none"> • Aceptar: botón que permite guardar los cambios. • Cancelar: botón que permite retornar a lista de los comentarios.
3.	Modifica los campos y selecciona la opción <i>Aceptar</i> .	<p>3.1 Valida que los campos.</p> <p>3.2 Muestra el mensaje “Su comentario ha sido enviado satisfactoriamente”.</p>
4.		4.1 Termina el caso de uso.
Flujos alternos		
3.1a “Existen campos vacíos”		
	Actor	Sistema
3.a		<p>3.1a Muestra el mensaje “<i>Campo en blanco</i>”.</p> <p>3.2a Retorna al punto 3.1 del flujo básico.</p>
3.1b “Formato incorrecto”		
3.b		<p>3.1b Muestra el mensaje: “Formato no válido”.</p> <p>3.2b Retorna al punto 3.1 del flujo básico.</p>
Flujos alternos		
2.1a “Cancelar”		
	Actor	Sistema
2.a	<i>Selecciona la opción Cancelar.</i>	2.1a Retorna a la interfaz donde se muestran todos los comentarios del recurso educativo. (Ver CU: <i>Ver comentarios de un</i>

		recurso educativo) 2.2a Termina el caso de uso.
Requisitos funcionales	RF15	

CU_13: Responder comentario de un recursos educativo.

Objetivo	Responder un comentario realizado a un recurso educativo.	
Actores	Usuario (inicia).	
Resumen	El caso de uso inicia cuando el usuario decide responder el comentario de un recurso educativo y finaliza cuando el comentario es respondido.	
Complejidad	Bajo	
Prioridad	Secundario	
Precondiciones	Debe haberse ejecutado el CU: Ver comentarios de un recurso educativo, el usuario debe estar autenticado y debe existir al menos un comentario de un recurso educativo en el sistema.	
Postcondiciones	Se respondió un comentario de un recurso educativo.	
Flujo de eventos		
Flujo básico: Responder comentario de un recursos educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Responder</i> .	<p>1.1. Muestra una interfaz donde se puede introducir:</p> <ul style="list-style-type: none"> • Título: campo de texto que permite insertar el título de la respuesta comentario a crear. • Descripción: campo de texto que permite insertar el contenido de la respuesta del comentario. <p>Además contiene los botones :</p> <ul style="list-style-type: none"> - Aceptar: botón que permite realizar la acción de adicionar la respuesta del

		comentario. - Cancelar: botón que permite cancelar la acción.
2.	Introduce los datos de la respuesta del comentario y posteriormente selecciona la opción <i>Aceptar</i> .	2.1. Valida que los campos. 2.2 Adiciona la respuesta del comentario. 2.3 Muestra el mensaje “ <i>El comentario sea respondido correctamente</i> ”.
3.		3.1 Termina el caso de uso.
Flujos alternos		
1.1a “Cancelar”		
	Actor	Sistema
1.a	Selecciona la opción <i>Cancelar</i> .	1.1a Retorna a la interfaz en donde se muestran los comentarios de un recurso educativo. (Ver CU: <i>Ver comentarios de un recurso educativo</i>) 1.2a Termina el caso de uso.
2.1a “Existen campos vacíos”		
	Actor	Sistema
2.a		2.1a Muestra el mensaje “ <i>Completa este campo</i> ”. 2.2a Retorna al punto 2.1 del flujo básico.
2.1b “Formato incorrecto”		
	Actor	Sistema
2.b		2.1b Muestra el mensaje: “ <i>Formato no válido</i> ”. 2.2b Retorna al punto 2.1 del flujo básico.
Requisitos funcionales	RF19	

CU_14: Filtrar comentario de un recurso educativo.

Objetivo	Filtrar comentario por fecha.
Actores	Usuario (inicia).
Resumen	El caso de uso inicia cuando el usuario desea filtrar los comentarios de un

	recurso educativo y finaliza cuando se filtran los comentarios.	
Complejidad	Baja.	
Prioridad	Secundario.	
Precondiciones	Debe haberse ejecutado el CU: Adicionar comentario a un recurso educativo y el usuario debe estar autenticado en el sistema.	
Postcondiciones	Se filtró el comentario según la fecha.	
Flujo de eventos		
Flujo básico: Filtrar comentarios de un recurso educativo		
	Actor	Sistema
1.	Selecciona la fecha y la opción Buscar	1.1 Muestra los comentarios.
2.		2.1 Termina el caso de uso.
Flujo alternativo		
1.1a “No existen comentarios”		
	Actor	Sistema
1.a		1.1a Muestra el mensaje: “No existen comentarios”.
Requisitos funcionales	RF17	

CU_15: Denunciar comentario de un recurso educativo.

Objetivo	Denunciar algún error en el contenido del recurso educativo.
Actores	Usuario (inicia).
Resumen	El caso de uso inicia una vez que el usuario accede a la opción: <i>Comentar</i> en la ficha de un recurso educativo. El sistema permite adicionar una denuncia a un comentario. El caso de uso finaliza una vez que el usuario ha adicionado una denuncia a algún comentario.
Complejidad	Bajo
Prioridad	Secundario
Precondiciones	El usuario debe estar autenticado y haberse ejecutado el CU: Ver ficha de un recurso educativo.

Postcondiciones	Se denunció un recurso educativo determinado.	
Flujo de eventos		
Flujo básico: Denunciar un recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Denunciar</i> .	<p>1.1. Muestra la interfaz donde se puede introducir:</p> <ul style="list-style-type: none"> • Título: campo de texto que permite insertar el título de la denuncia. • Descripción: campo de texto que permite insertar la descripción de la denuncia. <p>Además muestra las siguientes opciones:</p> <ul style="list-style-type: none"> - Aceptar: botón que permite enviar la denuncia. - Cancelar: botón que permite retornar lista de comentarios.
2.	Introduce los datos de la denuncia y selecciona la opción <i>Aceptar</i> .	<p>2.1. Valida que los campos.</p> <p>2.2 Adiciona la denuncia.</p> <p>2.3 Muestra un mensaje “Se ha adicionado la denuncia correctamente”.</p>
3.		3.1 Termina el caso de uso.
Flujos alternos		
1.1a “Cancelar”		
	Actor	Sistema
1.a	Selecciona la opción <i>Cancelar</i> .	<p>1.1 Retorna a la interfaz en donde se muestran los comentarios de un recurso educativo. (Ver CU: <i>Ver comentarios de un recurso educativo</i>)</p> <p>1.2a Termina el caso de uso.</p>
2.1a “Existen campos vacíos”		
	Actor	Sistema
2.a		2.1a Muestra un mensaje “Completa este campo”.

		2.2a Retorna al punto 2.1 del flujo básico.
2.1b “Formato incorrecto”		
	Actor	Sistema
2.b		2.1b Muestra el mensaje: “Formato no válido”. 2.2b Retorna al punto 2.1 del flujo básico.
Requisitos funcionales	RF18	

CU_16: Eliminar comentario de un recurso educativo.

Objetivo	Eliminar comentarios de un recurso educativo que se encuentra en el repositorio.	
Actores	Administrador (inicia).	
Resumen	El caso de uso inicia cuando el administrador muestra el listado de los comentarios de los recursos educativos y finaliza cuando es eliminado.	
Complejidad	Alta.	
Prioridad	Secundario.	
Precondiciones	El recurso educativo debe tener al menos un comentario	
Postcondiciones	Queda eliminado el comentario del recurso educativo.	
Flujo de eventos		
Flujo básico: Eliminar comentario de un recurso		
	Actor	Sistema
1.	Lista los comentarios.	1.1 Muestra una interfaz con el listado de los comentarios. Contiene además la opción: - Eliminar: botón que permite eliminar el comentario del recurso educativo.
2.	Selecciona el comentario y la opción <i>Eliminar</i> .	2.1 Muestra el mensaje de confirmación: ¿Está seguro que desea eliminar el elemento seleccionado? Además muestra las opciones:

		<p>Aceptar: botón que confirma el proceso de eliminar un comentario de un recurso educativo.</p> <p>Cancelar: botón que retorna al listado de los comentarios.</p>
3.	Selecciona la opción <i>Aceptar</i> .	<p>3.1 Elimina el comentario.</p> <p>3.2 Muestra el mensaje: “El elemento se ha eliminado correctamente”</p>
4.		4.1 Termina el caso de uso.
Flujos alternos		
2.1a “Cancelar”		
	Actor	Sistema
2.a	Selecciona opción <i>Cancelar</i> .	<p>2.1a Retorna a la vista donde se muestra el listado de los comentarios.</p> <p>2.2a Termina el caso de uso.</p>
Requisitos funcionales	RF16	

CU_17: Calificar recurso educativo.

Objetivo	Calificar un recurso educativo.	
Actores	Usuario (Inicia).	
Resumen	El caso de uso inicia una vez que el usuario se encuentra en la ficha de un recurso educativo permitiendo otorgar una valoración al mismo y finaliza una vez que es calificado el recurso educativo.	
Complejidad	Baja.	
Prioridad	Secundaria.	
Precondiciones	El usuario debe estar autenticado en el sistema y debe haberse ejecutado el CU: Ver ficha de un recurso educativo.	
Postcondiciones	Queda calificado el recurso educativo según el usuario.	
Flujo de eventos		
Flujo básico: Calificar recurso educativo		
	Actor	Sistema
1.	Selecciona la calificación	1.1 Permite calificar según el rating de estrellas.

	deseada.	
2.		2.1. Guarda calificación. 2.2. Se refleja en el rating de estrellas la calificación del recurso educativo.
3.		3.1. Termina el caso de uso.
Requisitos funcionales	RF22	

CU_18: Denunciar recurso educativo.

Objetivo	Denunciar algún error en el contenido del recurso educativo, plagio o violación de derechos de autor.	
Actores	Usuario (inicia).	
Resumen	El caso de uso inicia cuando el usuario selecciona la opción <i>Denunciar</i> y finaliza una vez que la denuncia es creada.	
Complejidad	Bajo	
Prioridad	Secundario	
Precondiciones	Debe haberse ejecutado el CU: Ver ficha del recurso educativo y además el usuario debe estar autenticado.	
Postcondiciones	Se denunció un recurso educativo.	
Flujo de eventos		
Flujo básico: Denunciar un recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Denunciar</i> .	1.1. Muestra la interfaz donde se puede introducir: <ul style="list-style-type: none"> • Título: campo de texto que permite insertar el título que tendrá la denuncia. • Descripción: campo de texto que permite insertar el contenido de la

		<p>denuncia.</p> <ul style="list-style-type: none"> • Imagen: campo de tipo file que permite localizar la dirección de la imagen que será incluida en la denuncia. (Opcional) • Así como los botones: <ul style="list-style-type: none"> - Aceptar: botón que permite realizar la acción de denunciar. - Cancelar: botón que permite cancelar la acción.
2.	Introduce los datos de la denuncia y selecciona la opción <i>Aceptar</i> .	<p>2.1. Valida que los campos.</p> <p>2.2. Almacena la denuncia realizada.</p> <p>2.3. Muestra el mensaje <i>“La denuncia se ha adicionado correctamente”</i>.</p>
3.		3.1 Termina el caso de uso.
Flujos alternos		
1.1a “Cancelar”		
	Actor	Sistema
1.a	Selecciona la opción <i>Cancelar</i> .	<p>1.1a Retorna a la interfaz de la ficha del recurso educativo. (Ver CU: <i>Ver ficha de un recurso educativo</i>)</p> <p>1.2a Termina el caso de uso.</p>
2.1a “Existen campos vacíos”		
	Actor	Sistema
2.a		<p>2.1a Muestra el mensaje <i>“Completa este campo”</i>.</p> <p>2.2a Retorna al punto 2.1 del flujo básico.</p>
2.1b “Formato incorrecto”		
	Actor	Sistema
2.b		2.1b Muestra el mensaje: <i>“Formato no</i>

		válido". 2.2b Retorna al punto 2.1 del flujo básico.
Requisitos funcionales	RF21	

CU_20: Adicionar sugerencias al autor de un recurso educativo.

Objetivo	Realizar una sugerencia al autor de un recurso educativo que se encuentre en el repositorio.	
Actores	Usuario (inicia).	
Resumen	El caso de uso inicia cuando el usuario decide realizar una sugerencia al autor del recurso educativo, el usuario introduce el contenido y es almacenada posteriormente en el sistema, finalizando así el caso de uso.	
Complejidad	Bajo.	
Prioridad	Secundario.	
Precondiciones	Debe estar autenticado en el sistema y debe haberse ejecutado el CU: Listar sugerencias de un recurso educativo.	
Postcondiciones	Se realizó la sugerencia al autor de un recurso educativo.	
Flujo de eventos		
Flujo básico: Adicionar sugerencia al autor de un recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Adicionar</i> .	1.1. Muestra una interfaz donde se puede introducir: <ul style="list-style-type: none"> • Título: campo de texto que permite insertar el título de la sugerencia a crear. • Descripción: campo de texto que permite insertar el contenido de la sugerencia. • Imagen: campo de tipo file que permite localizar la dirección de la

		<p>imagen que será incluida en la sugerencia. (Opcional)</p> <p>Además contiene las opciones :</p> <ul style="list-style-type: none"> - Aceptar: botón que permite realizar la acción de adicionar la sugerencia. - Cancelar: botón que permite regresar al listado de las sugerencias.
2.	Introduce los datos de la sugerencia y posteriormente selecciona la opción <i>Aceptar</i> .	<p>2.1 Valida que los campos.</p> <p>2.2 Adiciona la sugerencia.</p> <p>2.2 Muestra el mensaje “<i>La sugerencia se ha adicionado correctamente</i>”.</p>
3.		3.1 Termina el caso de uso.
Flujos alternos		
Evento 1.1a “Cancelar”		
	Actor	Sistema
1.a	Selecciona la opción <i>Cancelar</i> .	<p>1.1a Retorna a la lista de las sugerencias de un recurso educativo. (Ver CU: Listar sugerencias de un recurso educativo)</p> <p>1.2a Termina el caso de uso.</p>
Evento 2.1a “Existen campos vacíos”		
	Actor	Sistema
2.1a		<p>2.1a Muestra el mensaje “<i>Completa este campo</i>”.</p> <p>2.2a Retorna al punto 2.1 del flujo básico.</p>
Evento 2.1b “Formato incorrecto”		
	Actor	Sistema
2.b		<p>2.1b Muestra el mensaje: “<i>Formato no válido</i>”</p> <p>2.2b Retorna al punto 2.1 del flujo básico.</p>
Requisitos funcionales	RF23	

CU_21: Listar sugerencias de un recurso educativo.

Objetivo	Listar las sugerencias de un recurso educativo.	
Actores	Usuario (inicia).	
Resumen	El caso de uso se inicia cuando el usuario selecciona la opción <i>Sugerir</i> y finaliza cuando muestra el listado de las sugerencias del recurso educativo.	
Complejidad	Baja.	
Prioridad	Media.	
Precondiciones	Debe haberse ejecutado el CU: Ver ficha del recurso educativo y el usuario debe estar autenticado en el sistema.	
Postcondiciones	Muestra el listado de sugerencias del recurso educativo.	
Flujo de eventos		
Flujo básico : Listar sugerencias de un recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Sugerir</i> .	<p>1.1 Muestra una vista con el listado de sugerencias del recurso educativo. La misma contiene la siguiente información:</p> <ul style="list-style-type: none"> • Título: nombre de la sugerencia del recurso educativo. • Autor: nombre del autor de la sugerencia. • Fecha: fecha de creación de la sugerencia. • Acciones: muestra la opción: Mostrar Ver CU: <i>Ver sugerencia del recurso educativo</i>. <p>Además muestra la opción Adicionar: botón que permite adicionar una sugerencia al recurso educativo seleccionado. Ver CU: <i>Adicionar sugerencias al autor de un recurso educativo</i>.</p>
2.		2.1 Termina el caso de uso.

Requisitos funcionales	RF24
-------------------------------	------

CU_22: Listar sugerencias de todos recursos educativos.

Objetivo	Listar las sugerencias de todos los recursos educativos.	
Actores	Usuario (inicia).	
Resumen	El caso de uso se inicia cuando el usuario selecciona en el menú Recursos Educativos la opción <i>Mostrar sugerencias a recursos educativos</i> y finaliza cuando muestra el listado de las sugerencias de los recursos educativos.	
Complejidad	Baja.	
Prioridad	Media.	
Precondiciones	El usuario debe estar autenticado en el sistema.	
Postcondiciones	Muestra el listado de las sugerencias de los recursos educativos.	
Flujo de eventos		
Flujo básico: Listar sugerencias de un recurso educativo		
	Actor	Sistema
1.	Selecciona la opción <i>Mostrar sugerencias a recursos educativos</i> que se encuentra en el menú Recursos Educativos.	<p>1.1 Muestra una vista con el listado de las sugerencias de los recursos educativos. La misma contiene la siguiente información:</p> <ul style="list-style-type: none"> • Recurso Educativo: nombre del recurso educativo. • Autor: nombre del autor del recurso educativo. • Sugerencia: título de la sugerencia. • Acciones: muestra las opciones: Mostrar Ver CU: <i>Ver sugerencia del recurso educativo.</i> Adicionar Ver CU: <i>Adicionar sugerencias al autor de un recurso educativo.</i>
2.		2.1 Termina el caso de uso.

Requisitos funcionales	RF25
-------------------------------	------

CU_23: Ver sugerencia de un recurso educativo.

Objetivo	Mostrar los detalles de la sugerencia de un recurso educativo.
Actores	Usuario (inicia).
Resumen	El caso de uso inicia cuando el usuario decide ver una sugerencia y finalizando cuando muestra los detalles de la sugerencia.
Complejidad	Bajo
Prioridad	Secundario
Precondiciones	Debe haberse ejecutado el CU: <i>Listar sugerencias de un recurso educativo</i> o <i>Listar sugerencias de todos recursos educativos</i> y el usuario debe estar autenticado en el sistema.
Postcondiciones	Se muestra los detalles de la sugerencia

Flujo de eventos

Flujo básico: Ver sugerencia del recurso educativo

	Actor	Sistema
1.	Selecciona la opción <i>Mostrar</i> .	<p>1.1. Muestra una vista con la sugerencia del recurso educativo. La misma contiene la siguiente información:</p> <ul style="list-style-type: none"> • Título: muestra el título de la sugerencia. • Autor: muestra el nombre del autor de la sugerencia. • Fecha de creación: muestra la fecha de creación de la sugerencia. • Descripción: muestra el contenido de la sugerencia. <p>Además contiene el botón atrás: retorna a la vista de la ficha del recurso educativo.</p>

2.		2.1 Termina el caso de uso.
Requisitos funcionales	RF26	

CU_24: Eliminar sugerencias al autor de un recurso educativo.

Objetivo	Eliminar sugerencias a un recurso educativo que se encuentra en el repositorio.	
Actores	Administrador (inicia).	
Resumen	El caso de uso inicia cuando el administrador muestra el listado de las sugerencias de los recursos educativos y finaliza cuando es eliminada.	
Complejidad	Alta.	
Prioridad	Secundario.	
Precondiciones	El autor de un recurso debe haber recibido al menos una sugerencia de un recurso educativo de su autoría.	
Postcondiciones	Queda eliminada la sugerencia al autor del recurso educativo.	
Flujo de eventos		
Flujo básico: Eliminar sugerencias al autor de un recurso		
	Actor	Sistema
2.	Lista las sugerencias.	2.1 Muestra una interfaz con el listado de las sugerencias. Contiene además la opción: - Eliminar: botón que permite eliminar la sugerencia del recurso educativo.
2.	Selecciona la sugerencia y la opción <i>Eliminar</i> .	2.1 Muestra el mensaje de confirmación: ¿Está seguro que desea eliminar el elemento seleccionado? Además muestra las opciones: Aceptar: botón que permite confirmar el proceso de eliminar la sugerencia de un recurso educativo. Cancelar: botón que retorna al listado de las sugerencias.
3.		3.1 Elimina la sugerencia. 3.2 Muestra el mensaje: “El elemento se ha eliminado correctamente”

4.		4.1 Termina el caso de uso.
Flujos alternos		
2.1a "Cancelar"		
	Actor	Sistema
2.a	Selecciona opción <i>Cancelar</i> .	2.1a Retorna a la vista donde se muestra el listado de las sugerencias. 2.2a Termina el caso de uso.
Requisitos funcionales	RF27	

CU_25: Compartir recurso educativo en redes sociales.

Objetivo	Compartir en las redes sociales facebook y twitter el recurso educativo.	
Actores	Usuario	
Resumen	Cuando accedes a la vista Ver ficha del recurso educativo, la misma te brinda la posibilidad de compartirlo con facebook y twitter, el caso de uso finaliza cuando el recurso es compartido.	
Complejidad	Bajo	
Prioridad	Secundario	
Precondiciones	El usuario debe estar autenticado y debe haberse ejecutado el CU: Ver ficha del recurso educativo.	
Postcondiciones	Se envió la información del recurso educativo para ser compartido en las redes sociales.	
Flujo de eventos		
Flujo básico Compartir recurso educativo en redes sociales		
Sección 1: "Compartir en Twitter"		
	Actor	Sistema
1.	Selecciona la opción Compartir con twitter.	1.1 Crea un enlace con la red social twitter.
2.		1.1 Termina el caso de uso.
Flujos alternos		
1.1a No existe la conexión		
	Actor	Sistema
1.a		1.1a Muestra el mensaje: "No se puedo

		establecer la conexión con twitter”
Sección 2: “Compartir en Facebook”		
	Actor	Sistema
1.	Selecciona la opción Compartir con Facebook.	1.1 Crea un enlace con la red social facebook.
2.		2.1 Termina el caso de uso.
Flujos alternos		
1.1a No existe la conexión		
	Actor	Sistema
1.a		1.1a Muestra el mensaje: “No se puedo establecer la conexión con facebook”
Requisitos funcionales	RF28, RF29	

CU_26: Adicionar recurso educativo a favoritos.

Objetivo	Adicionar recurso educativo al listado de favoritos.	
Actores	Usuario (inicia)	
Resumen	El caso de uso inicia una vez que el usuario selecciona la opción adicionar a favoritos y finaliza cuando el recurso es adicionado a favoritos.	
Complejidad	Baja	
Prioridad	Secundario	
Precondiciones	Que se haya ejecutado el CU: Ver ficha del recurso educativo.	
Postcondiciones	Se ha adicionado el recurso educativo al listado de favoritos.	
Flujo de eventos		
Flujo básico: Adicionar recurso educativo a favoritos		
	Actor	Sistema
1	Selecciona la opción Adicionar a favoritos.	1.1 Muestra el mensaje: “El recurso ha sido adicionado a favoritos”.
2		2.1 Termina el caso de uso.

Flujo alternativo		
1a “El recurso existe en favoritos”		
	Actor	Sistema
1.a		1.1a Muestra el mensaje: “El recurso ya se encuentra en favoritos”.
Requisitos funcionales	RF30	

CU_28: Listar recursos educativos favoritos.

Objetivo	Mostrar el listado de los recursos educativos favoritos.	
Actores	Usuario (inicia)	
Resumen	El caso de uso inicia una vez que selecciona la opción Recursos favoritos y finaliza cuando muestra el listado de los recursos educativos favoritos.	
Complejidad	Baja.	
Prioridad	Baja.	
Precondiciones	Que el usuario esté autenticado en el sistema.	
Postcondiciones	Se muestra el listado de los recursos educativos favoritos.	
Flujo de eventos		
Flujo básico: Listar recursos educativos favoritos		
	Actor	Sistema
1.	Selecciona la opción Recursos favoritos.	<p>1.1 Muestra la vista de los recursos educativos favoritos. La misma contiene los siguiente información:</p> <ul style="list-style-type: none"> • Título: nombre del recurso educativo. • Autor: nombre del autor del recurso educativo. • Fecha: fecha de publicación del recurso educativo. • Acciones: muestra la opción Mostrar (<i>Ver CU: Ver ficha del recurso</i>

		<i>educativo).</i>
2.		2.1 Termina el caso de uso.
Requisitos funcionales	RF31	

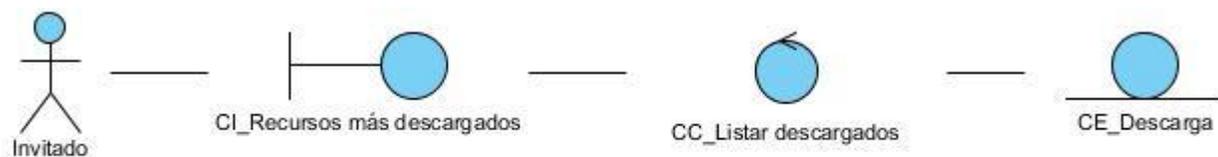
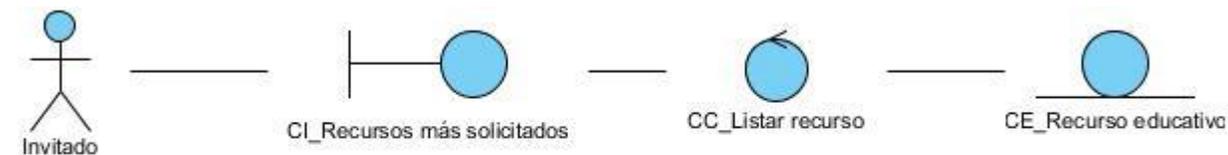
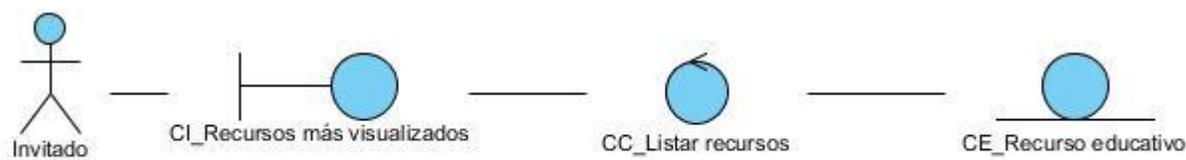
CU_29: Listar publicaciones.

Objetivo	Listar los recursos educativos de su autoría.	
Actores	Autor (inicia).	
Resumen	El caso de uso se inicia cuando el autor selecciona la opción de <i>Mis publicaciones</i> y finaliza cuando muestra el listado de recursos educativos de su autoría.	
Complejidad	Baja.	
Prioridad	Media.	
Precondiciones	El autor debe estar autenticado en el sistema.	
Postcondiciones	Muestra el listado de recursos educativos de su autoría.	
Flujo de eventos		
Flujo básico Listar publicaciones		
	Actor	Sistema
1.	Selecciona la opción <i>Mis publicaciones.</i>	<p>1.1 Muestra la vista con los recursos de su autoría.</p> <p>La misma contiene la siguiente información:</p> <ul style="list-style-type: none"> • Título: nombre del recurso educativo. • Colección: nombre de la colección a la que pertenece el recurso educativo. • Fecha de publicación: fecha en la que fue publicado el recurso educativo. • Acciones: muestra las opciones : <p>Mostrar Ver CU: <i>Ver ficha del recurso educativo.</i></p> <p>Editar Ver CU: <i>Editar las propiedades de un recurso educativo.</i></p>

2.		2.1 Termina el caso de uso.
Requisitos funcionales	RF32	

Anexo II. Diagramas de clases del análisis

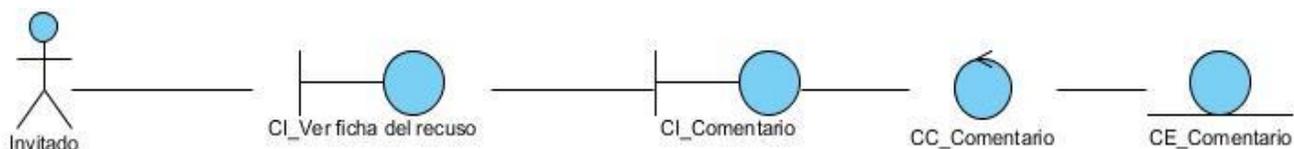
DCA_1: Listar recursos educativos por categoría (Más recientes, Más visualizados, Más descargados, Más solicitados).



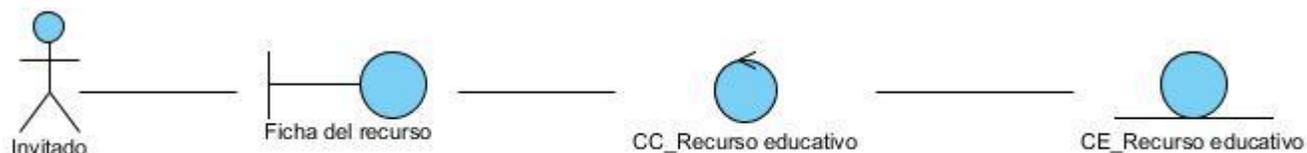
DCA_2: Exportar recurso educativo.



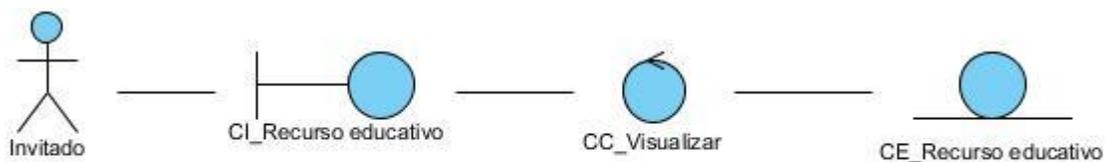
DCA _3: Ver comentarios de un recurso educativo.



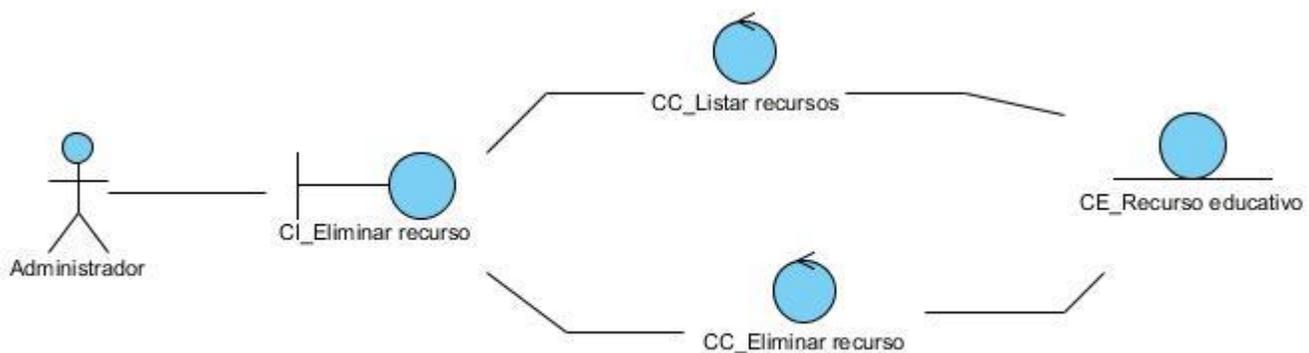
DCA _4: Ver ficha de un recurso educativo.



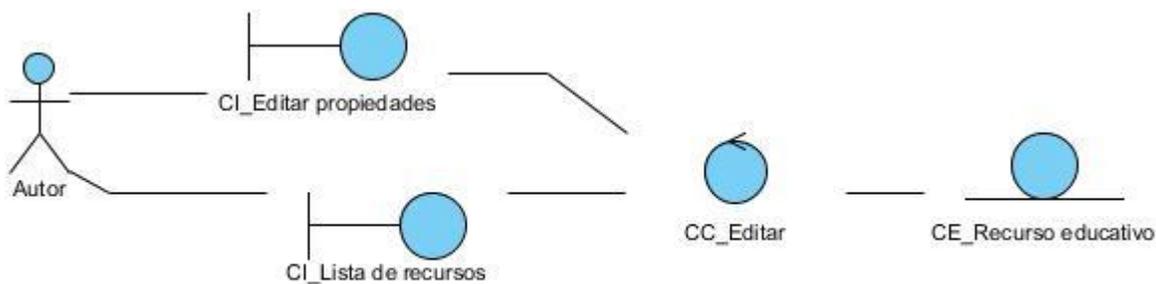
DCA _5: Visualizar recurso educativo.



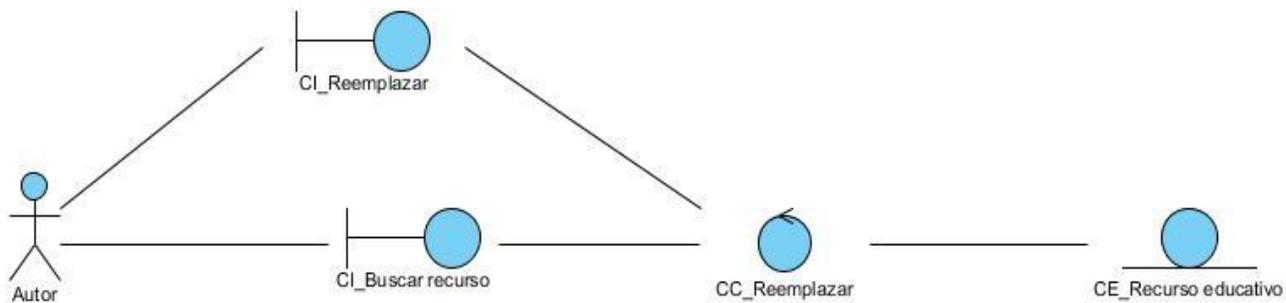
DCA _6: Eliminar recurso educativo.



DCA _7: Editar las propiedades de un recurso educativo.



DCA _8: Reemplazar recurso educativo.



DCA _9: Adicionar comentario a un recurso educativo.



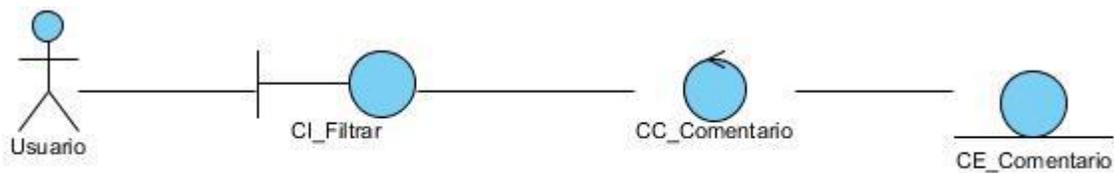
DCA _10: Editar comentario de un recurso educativo.



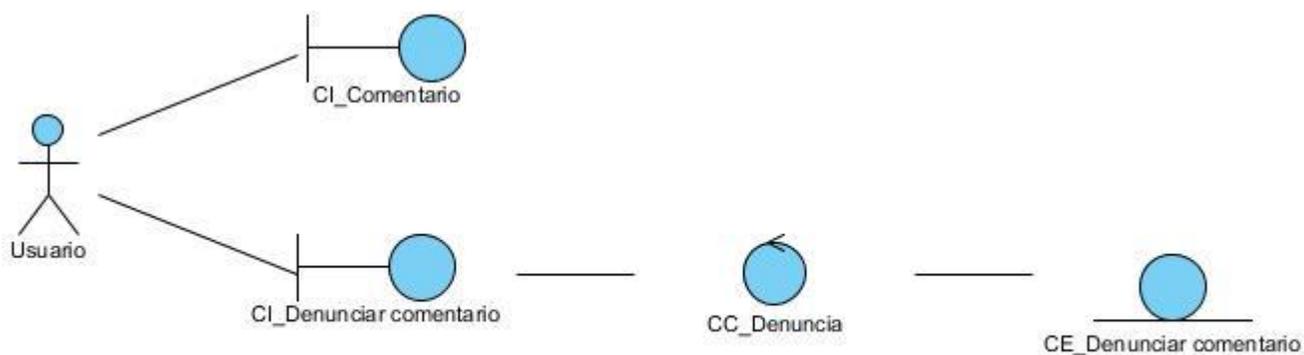
DCA _11: Responder comentario de un recurso educativo.



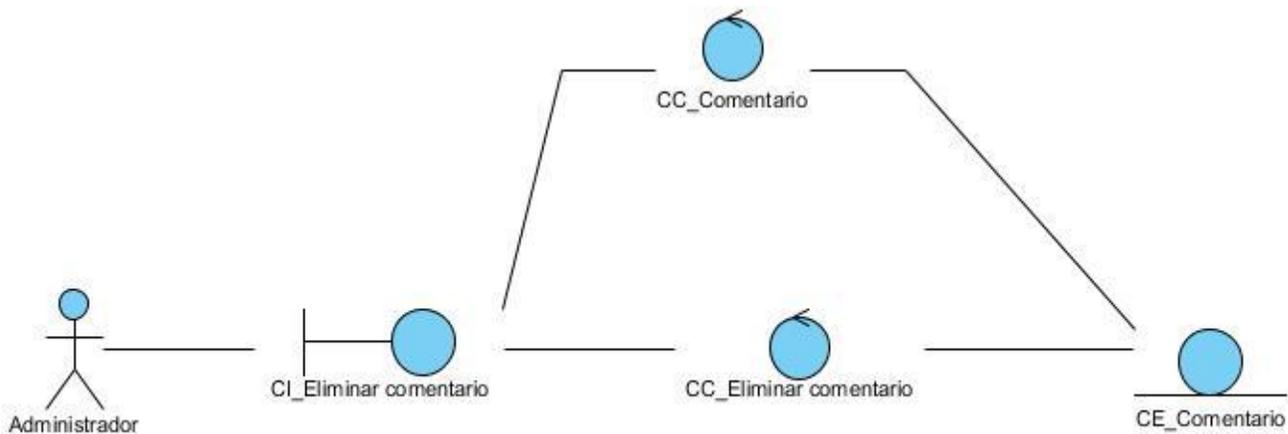
DCA _12: Filtrar comentario de un recurso educativo.



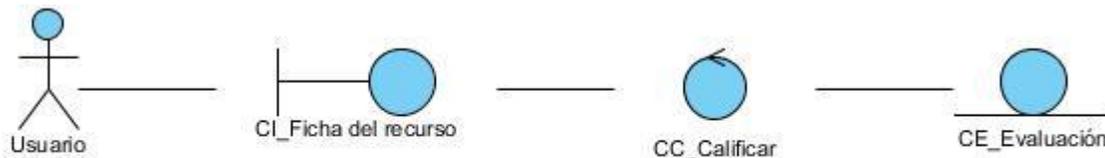
DCA _13: Denunciar comentario de un recurso educativo.



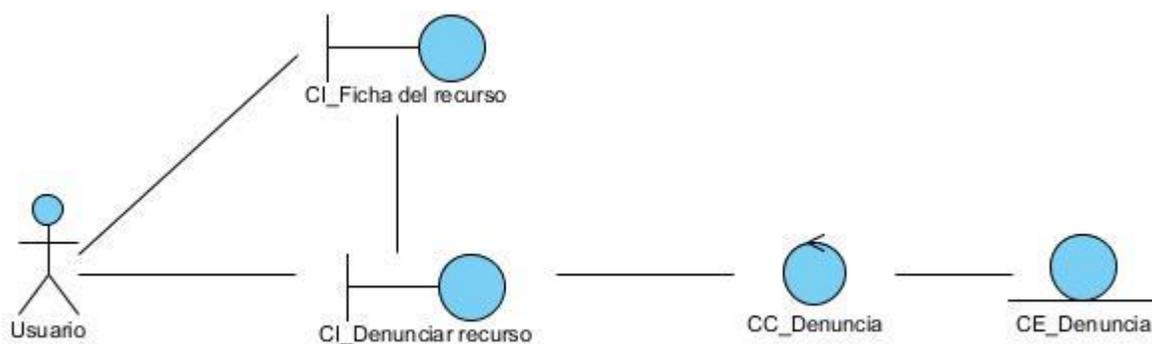
DCA _16: Eliminar comentario de un recurso educativo.



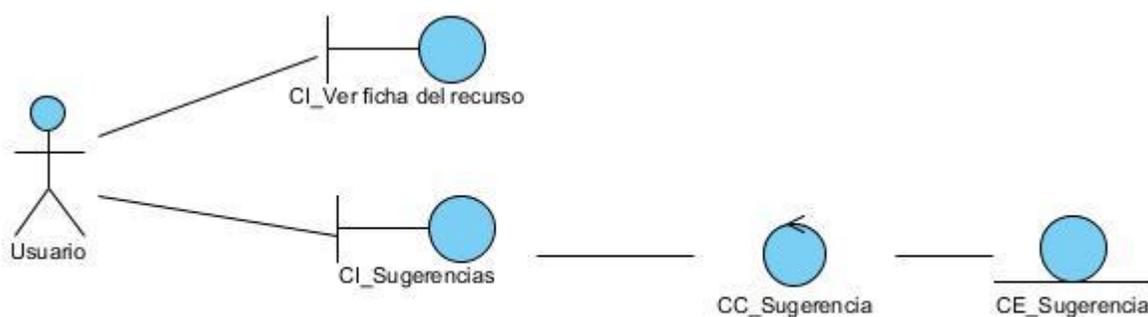
DCA _17: Calificar recurso educativo.



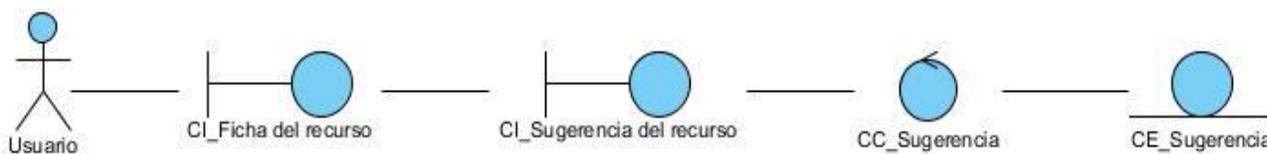
DCA _18: Denunciar recurso educativo.



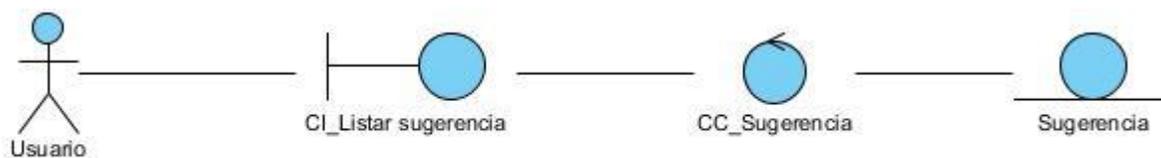
DCA _20: Adicionar sugerencias al autor de un recurso educativo.



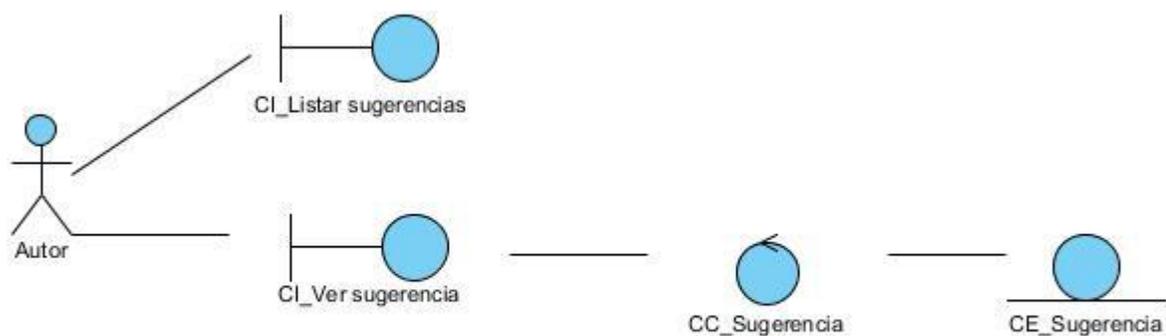
DCA _21: Listar sugerencias de un recurso educativo.



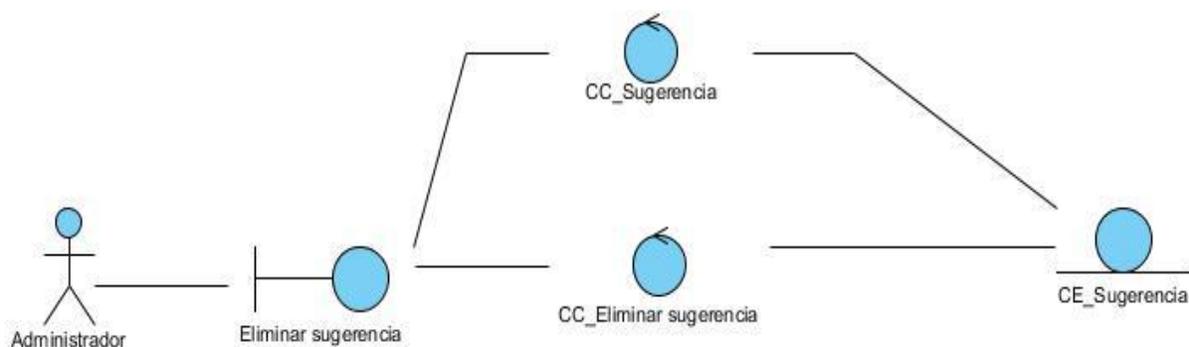
DCA _22: Listar sugerencias de todos recursos educativos.



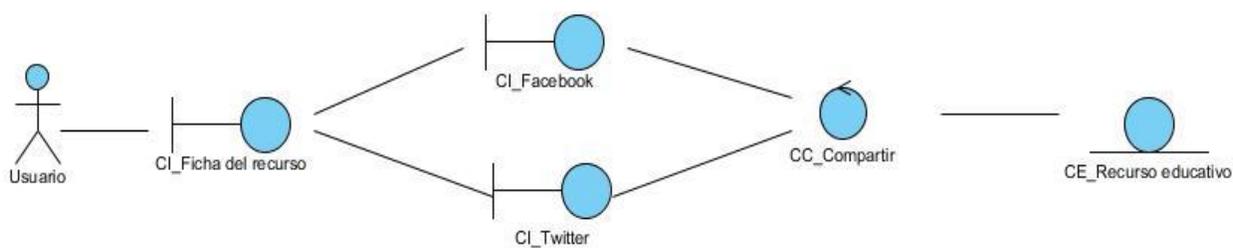
DCA _23: Ver sugerencia de un recurso educativo.



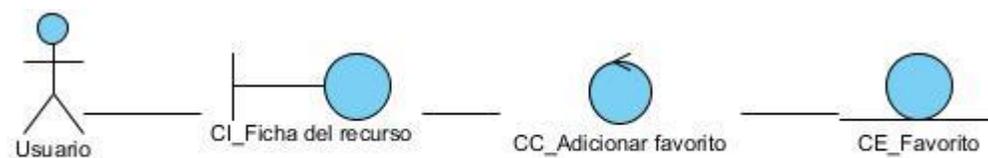
DCA_24: Eliminar sugerencias al autor de un recurso educativo.



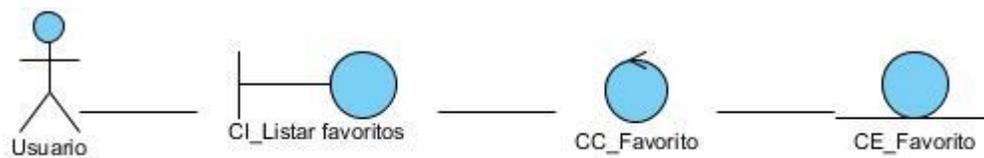
DCA_25: Compartir recurso educativo en redes sociales.



DCA_26: Adicionar recurso educativo a favoritos.



DCA _27: Listar recursos educativos favoritos.



DCA _28: Listar publicaciones.



Anexo III. Diagramas de colaboración

DC_1: Listar recursos educativos por categoría:(Más recientes, Más visualizados, Más descargados, Más solicitados).

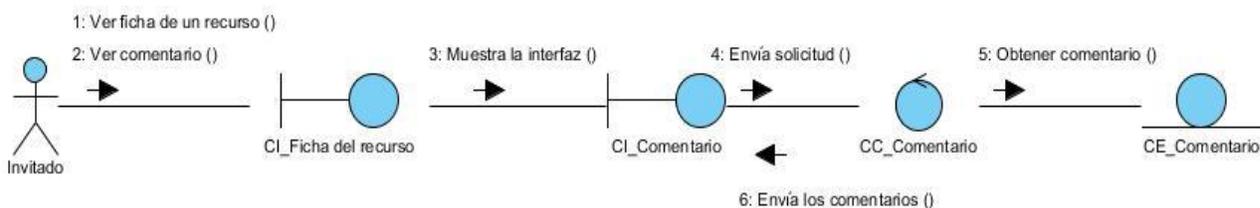




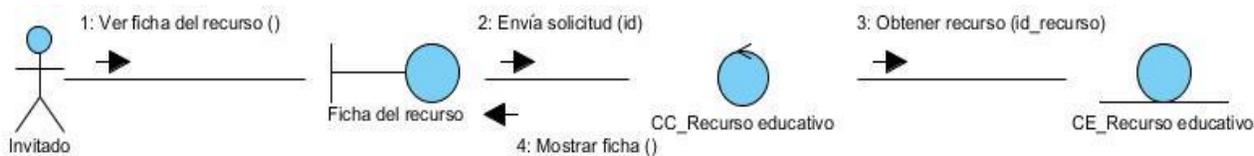
DC_2: Exportar recurso educativo.



DC_3: Ver comentarios de un recurso educativo.



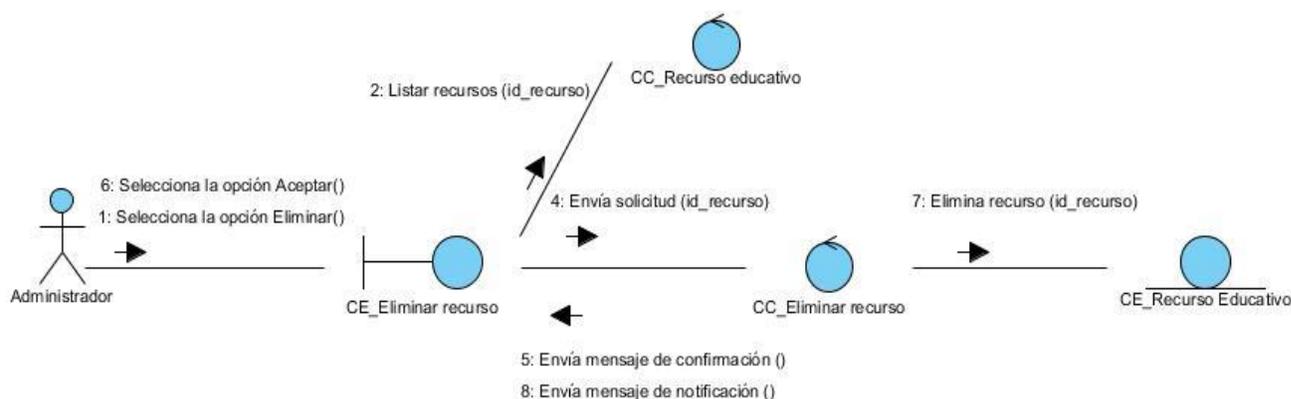
DC_6: Ver ficha de un recurso educativo.



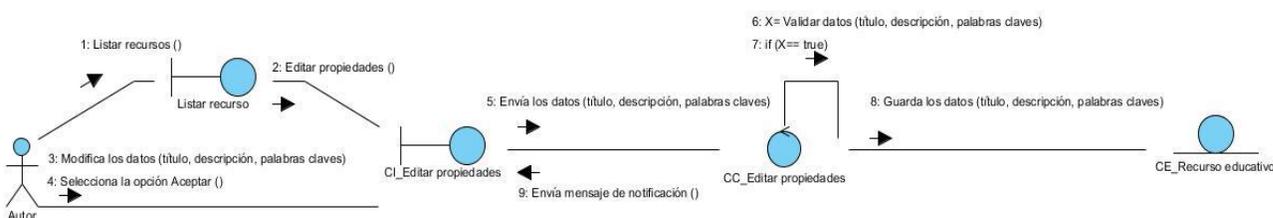
DC_7: Visualizar recurso educativo.



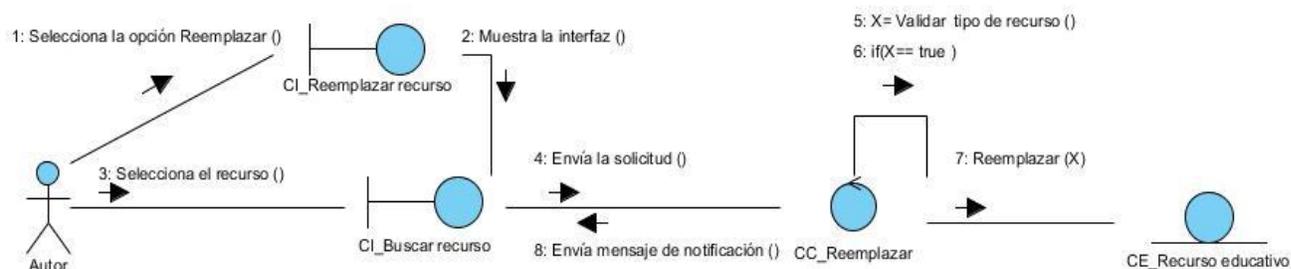
DC_8: Eliminar recurso educativo.



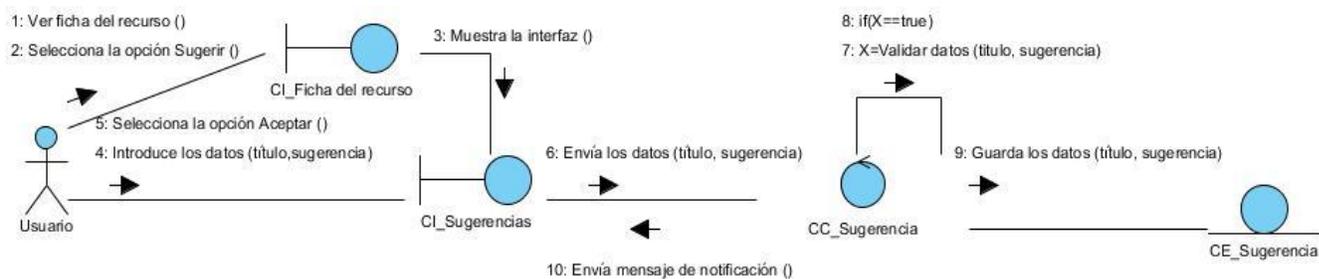
DC_9: Editar las propiedades de un recurso educativo.



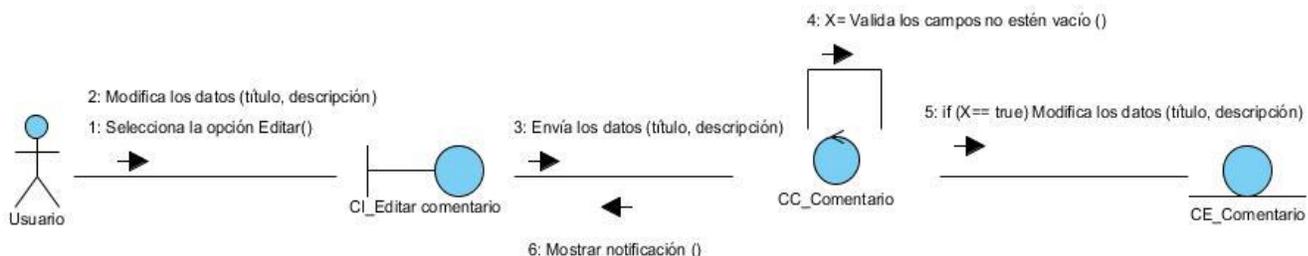
DC_10: Reemplazar recurso educativo.



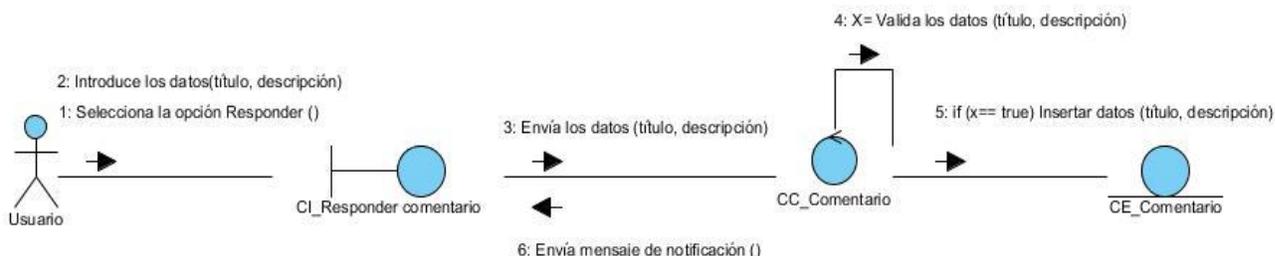
DC_11: Adicionar comentario a un recurso educativo.



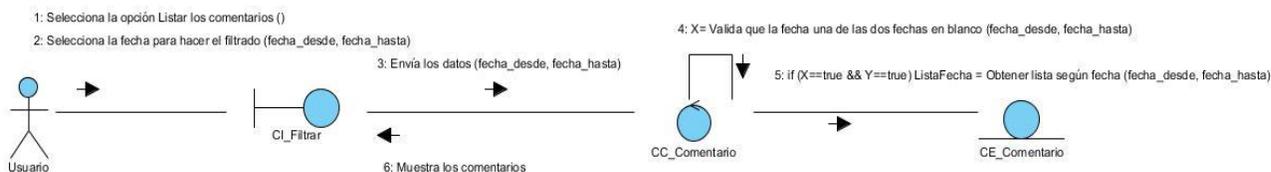
DC_12: Editar comentario de un recurso educativo.



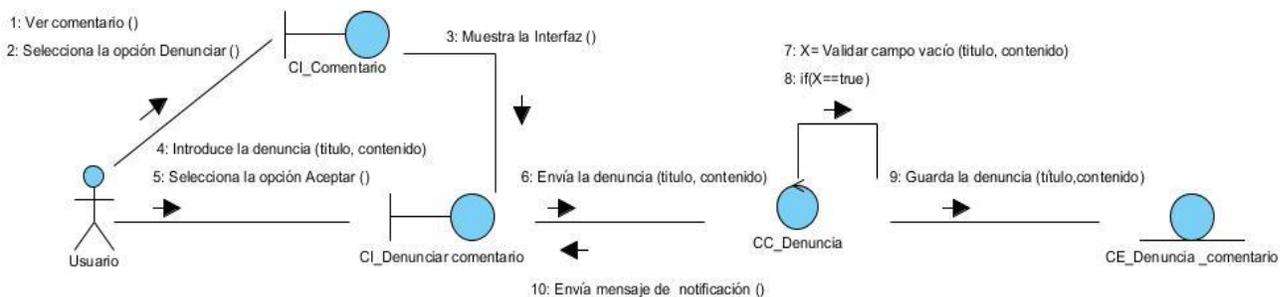
DC_13: Responder comentario de un recurso educativo.



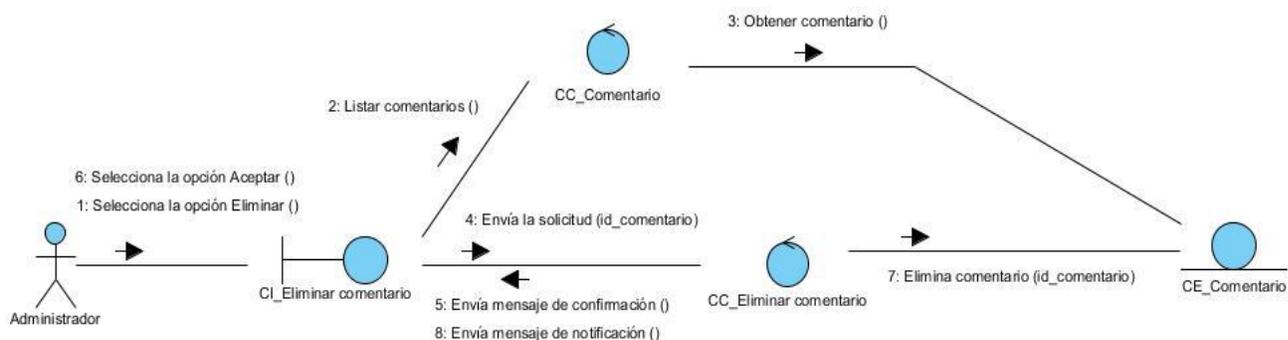
DC_14: Filtrar comentario de un recurso educativo.



DC_15: Denunciar comentario de un recurso educativo.



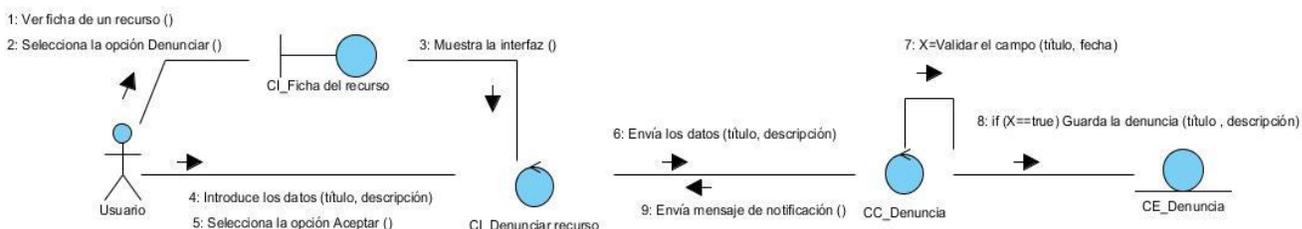
DC_16: Eliminar comentario de un recurso educativo.



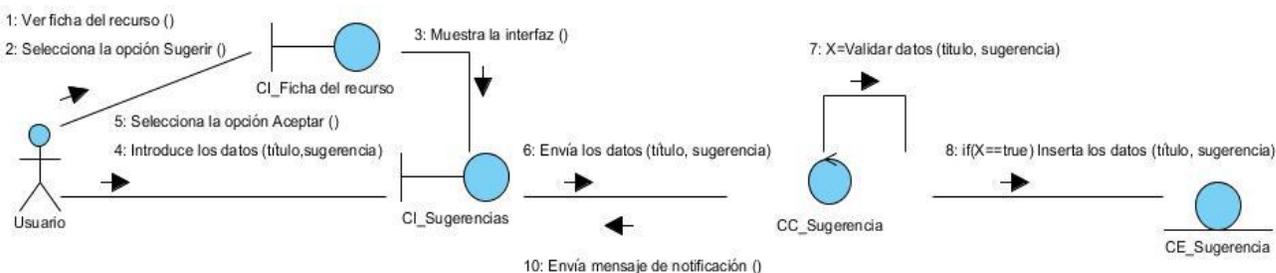
DC _17: Calificar recurso educativo.



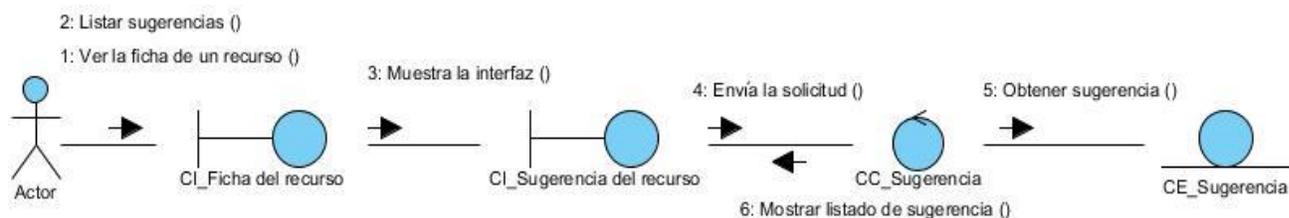
DC _18: Denunciar recurso educativo.



DC _19: Adicionar sugerencia al autor de un recurso educativo.



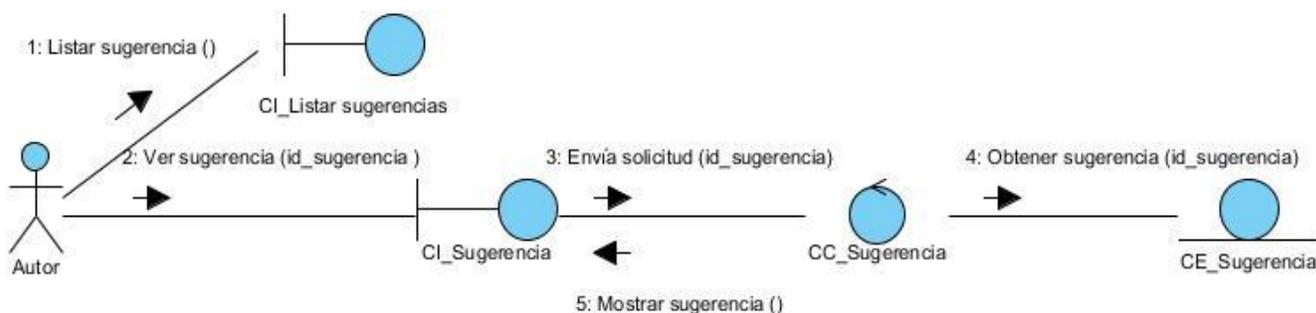
DC _21: Listar sugerencias de un recurso educativo.



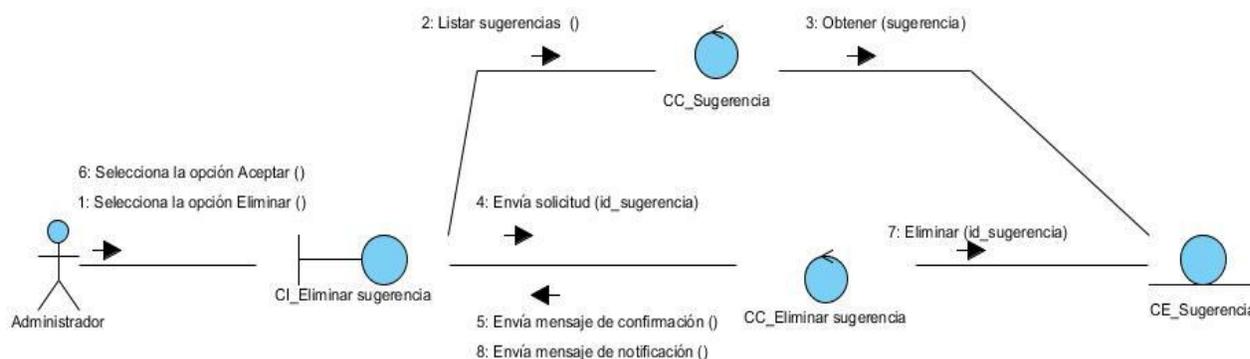
DC _22: Listar sugerencias de todos recursos educativos.



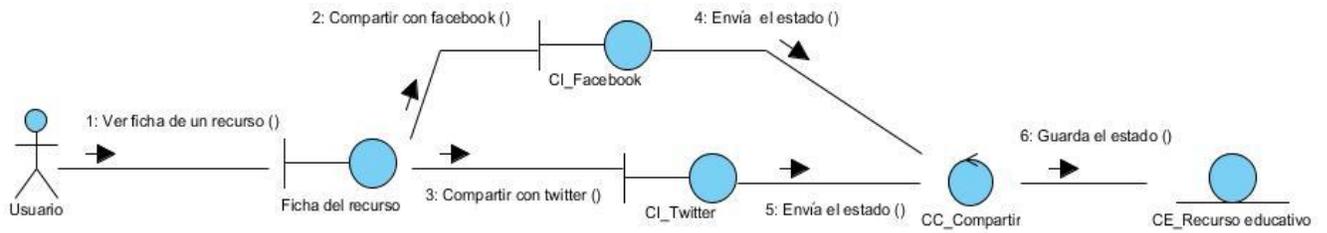
DC _23: Ver sugerencia de un recurso educativo.



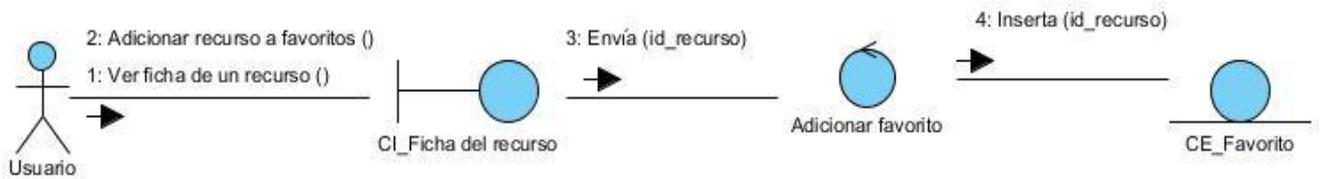
DC _24: Eliminar sugerencias al autor de un recurso educativo.



DC _25: Compartir recurso educativo en redes sociales.



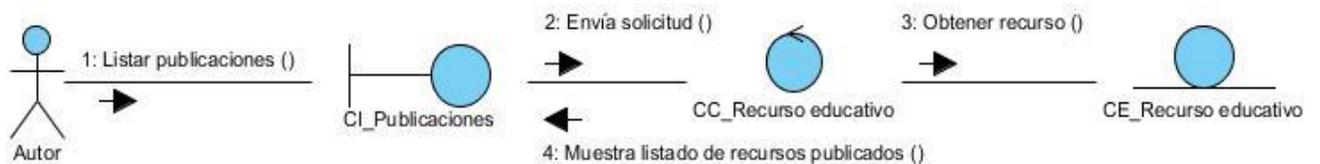
DC _26: Adicionar recurso educativo a favoritos.



DC _27: Listar recursos educativos favoritos.

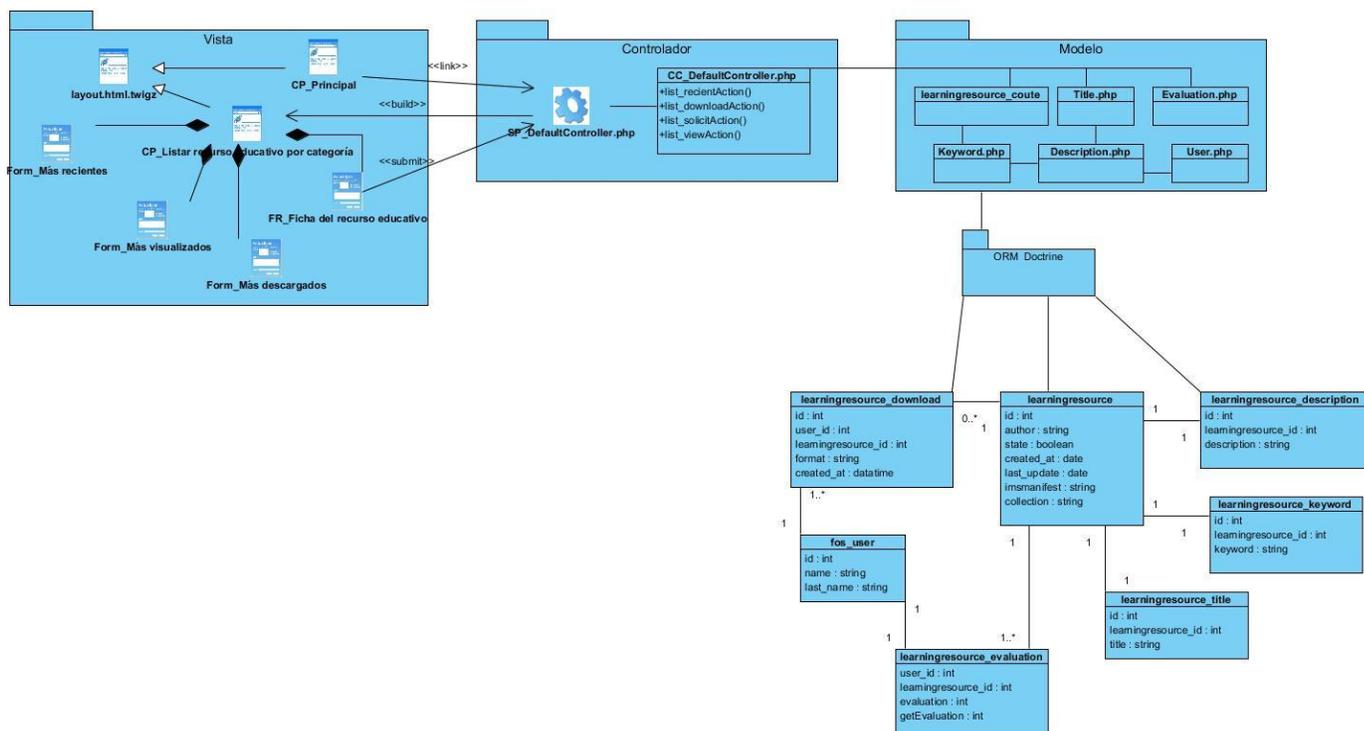


DC _28: Listar publicaciones.

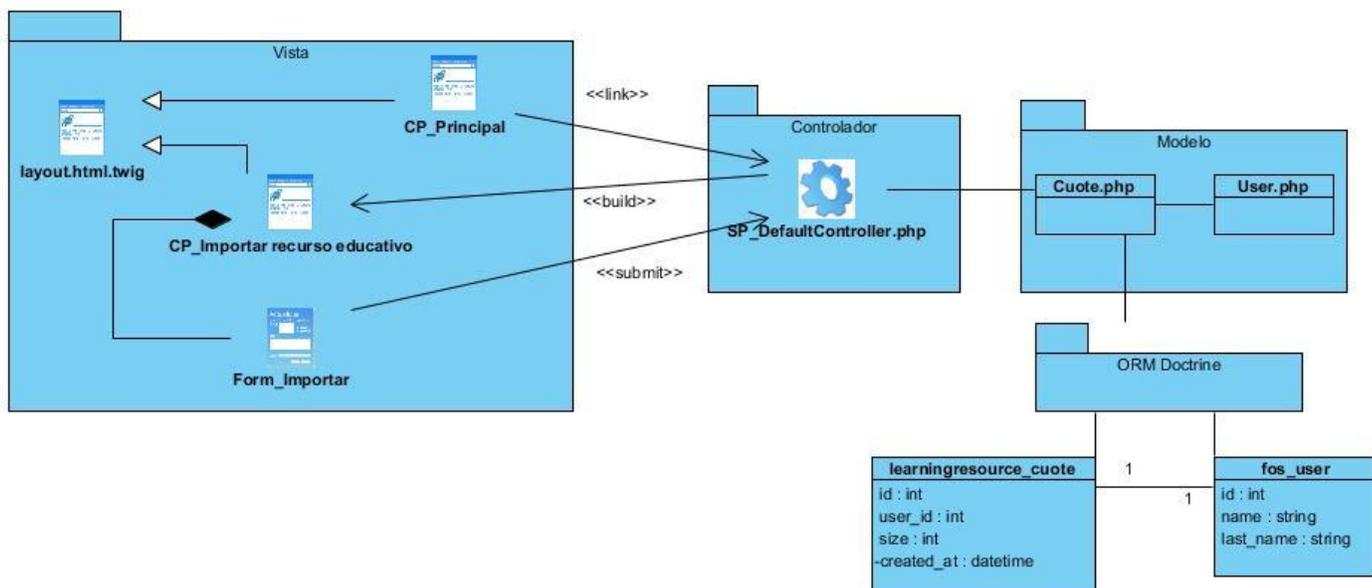


Anexo IV. Diagramas de clases del diseño

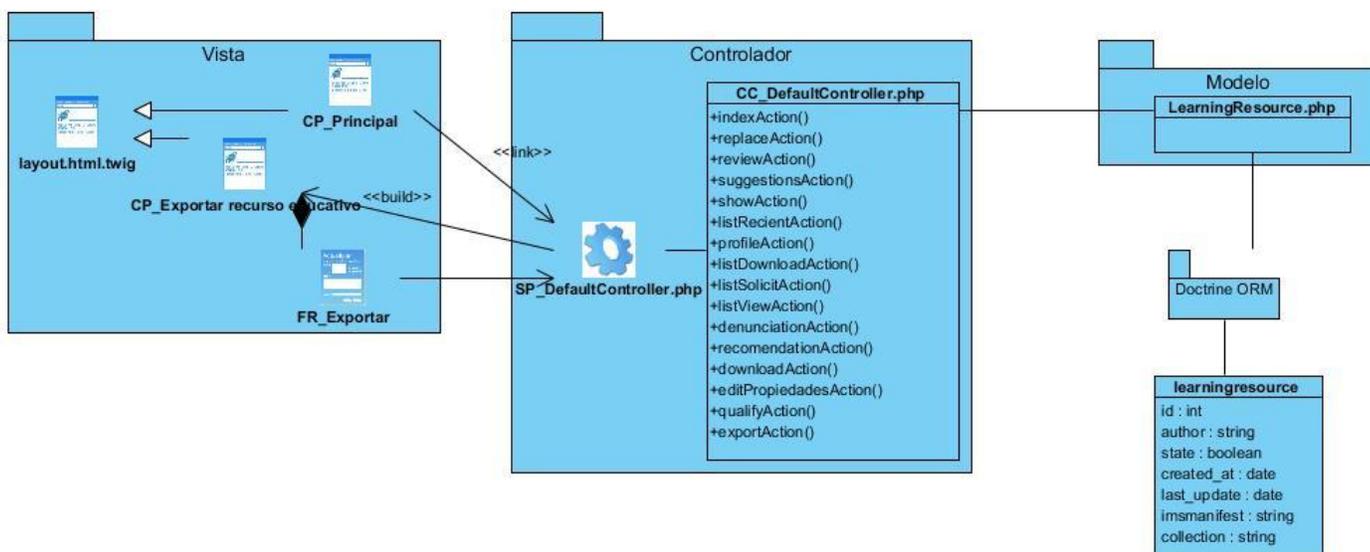
DCD_1: Listar los recursos educativos por categoría. Criterios (Más visualizados, más recientes, más solicitados, más descargados)



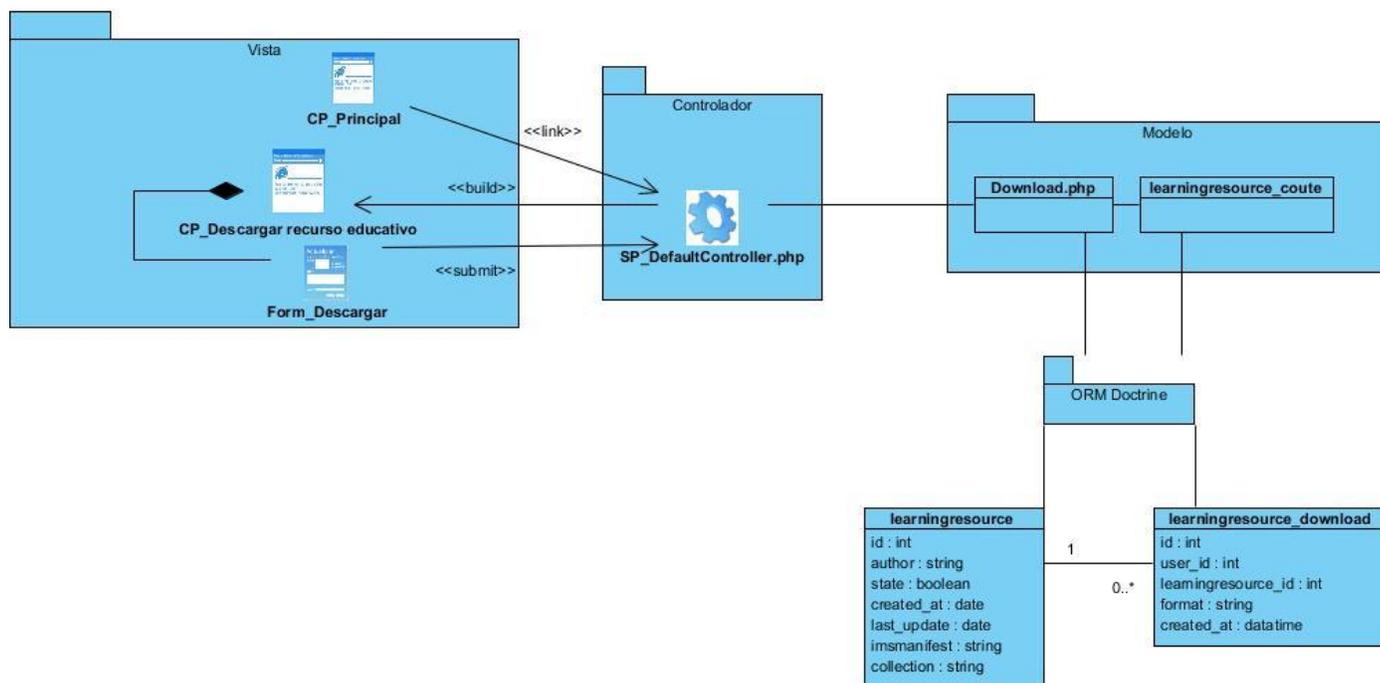
DCD_2: Importar recurso educativo.



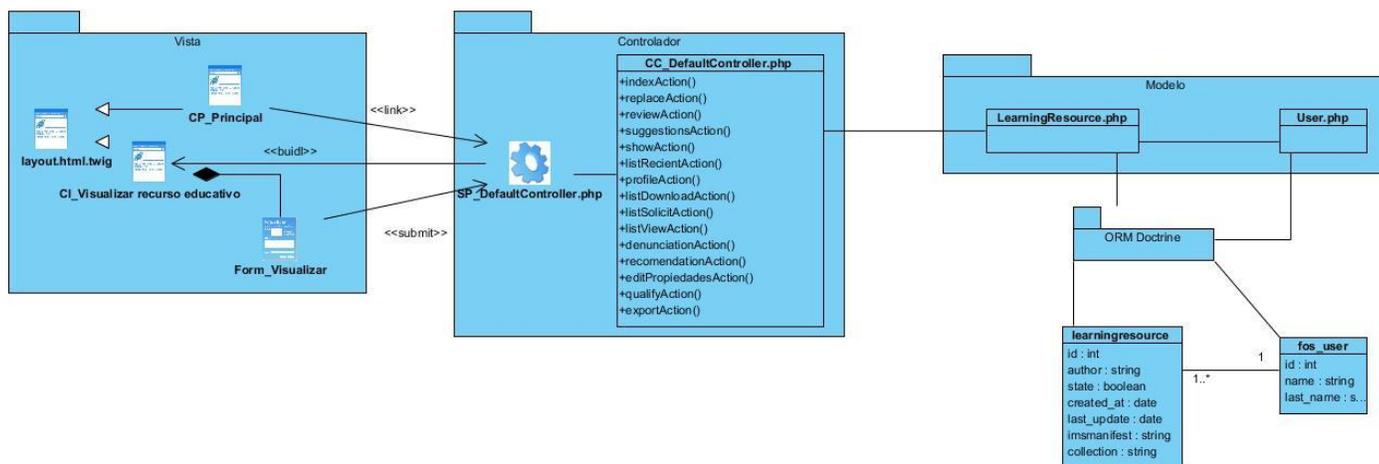
DCD_6: Exportar recurso educativo.



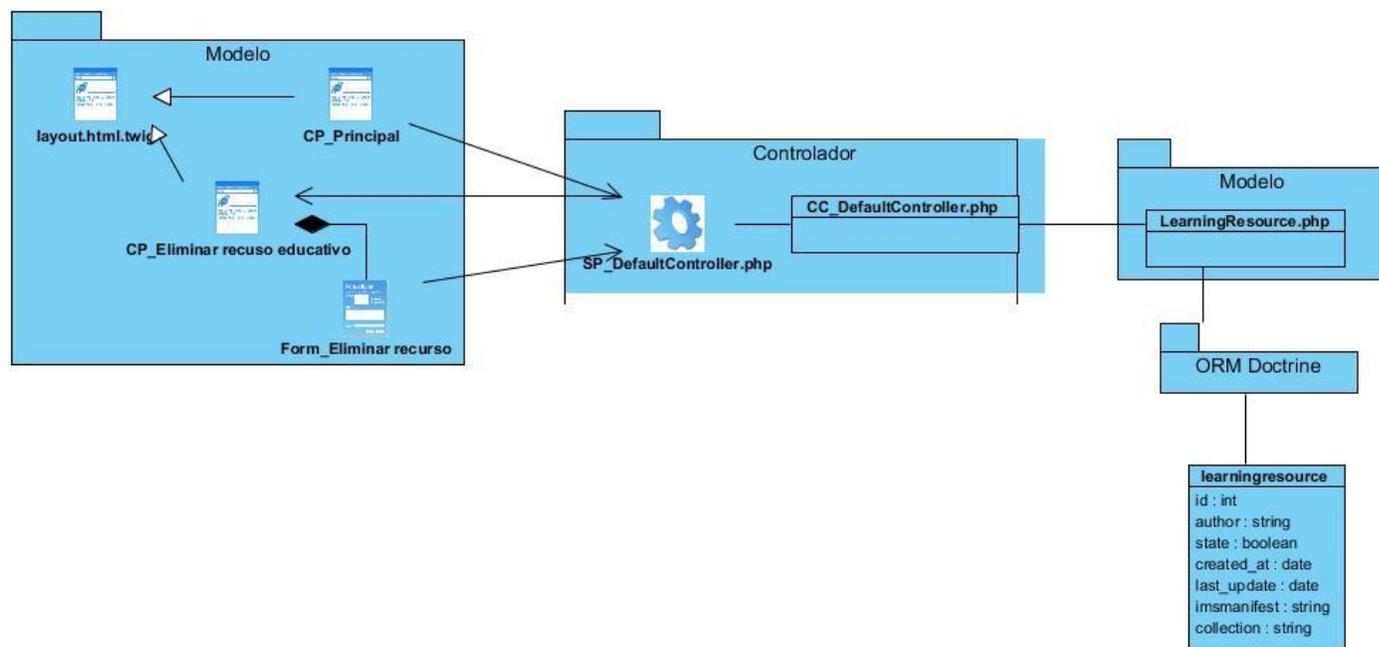
DCD_7: Descargar recurso educativo.



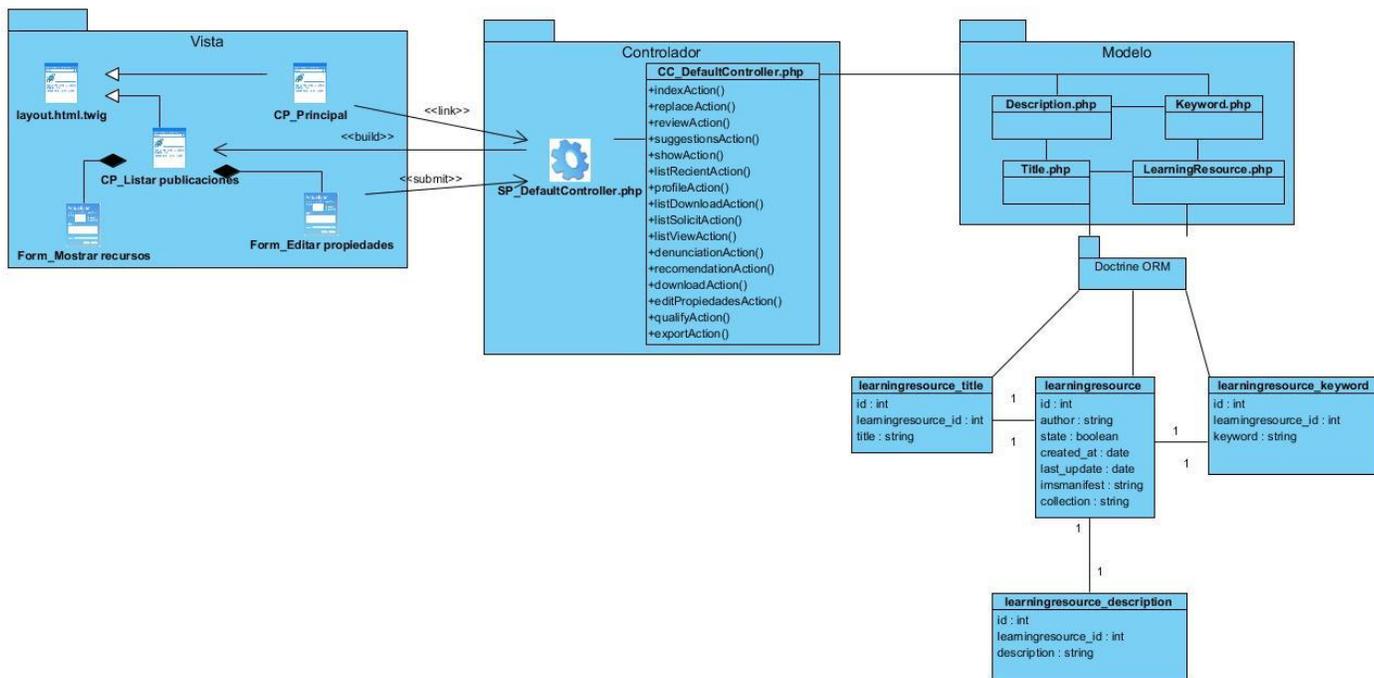
DCD_8: Visualizar recurso educativo.



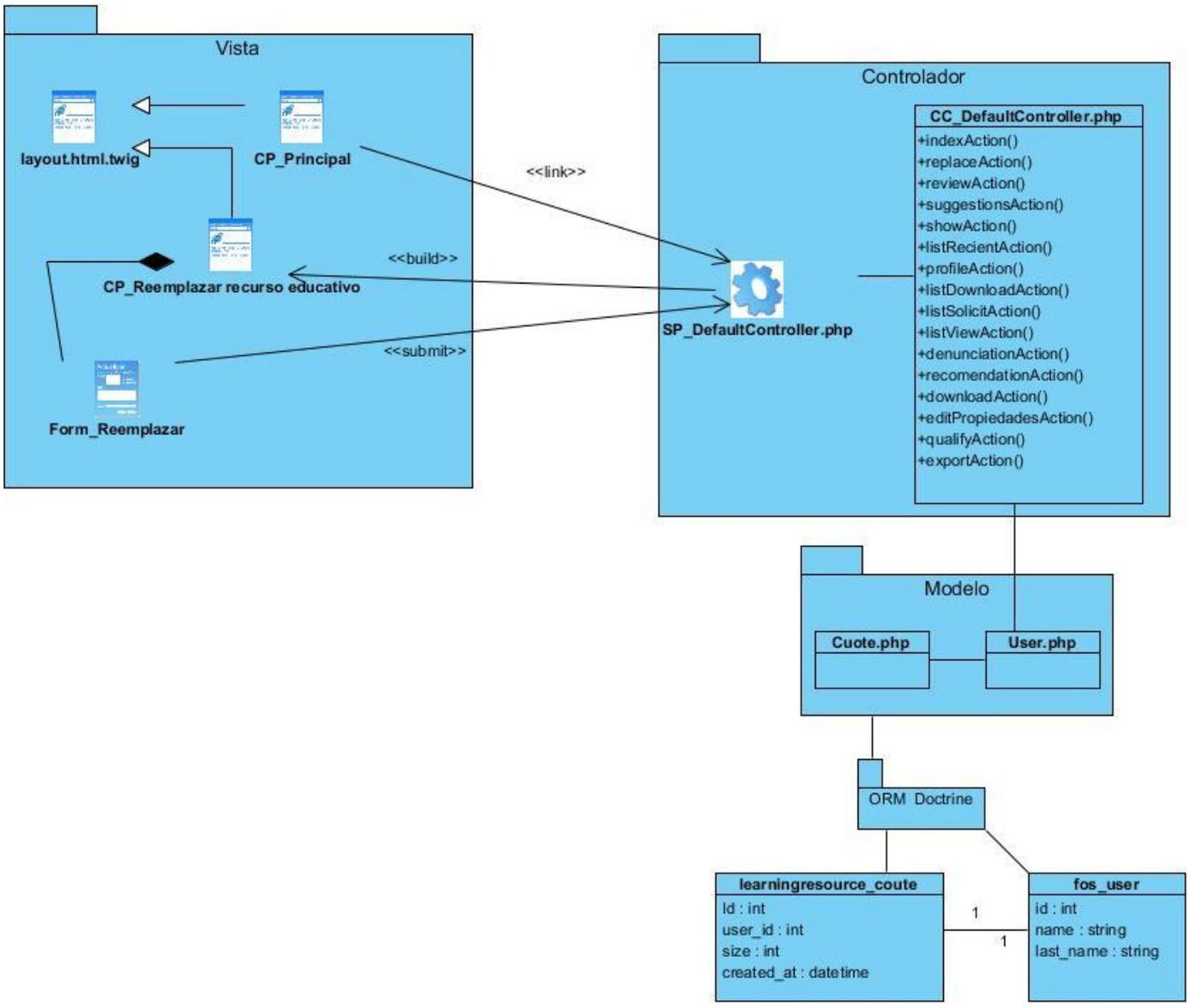
DCD_9: Eliminar recurso educativo.



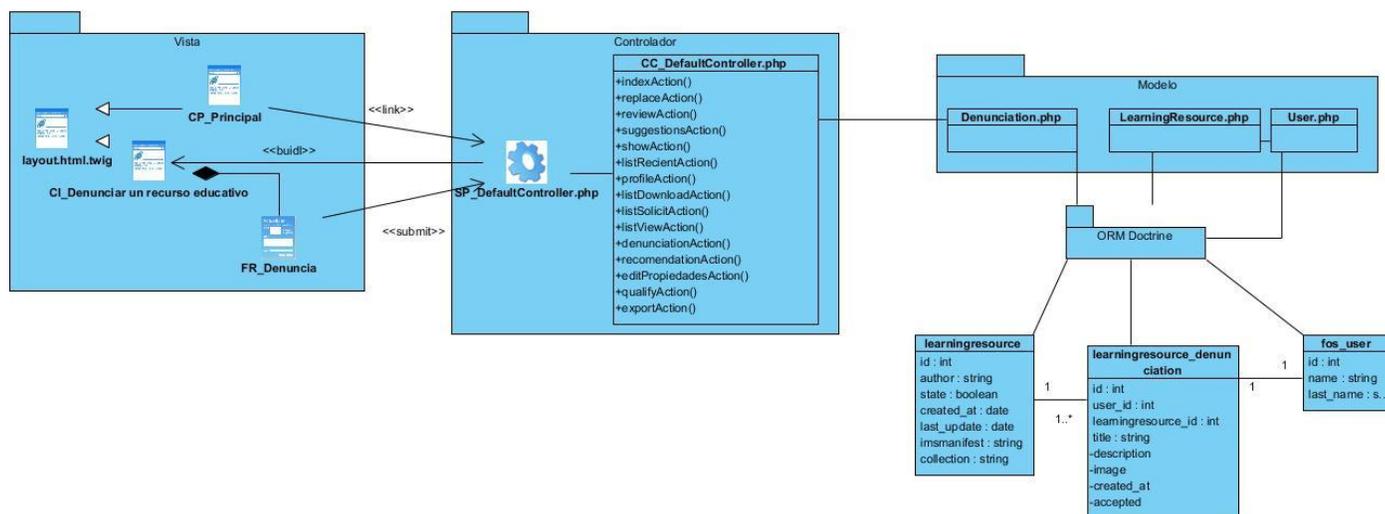
DCD_10: Editar las propiedades de un recurso educativo.



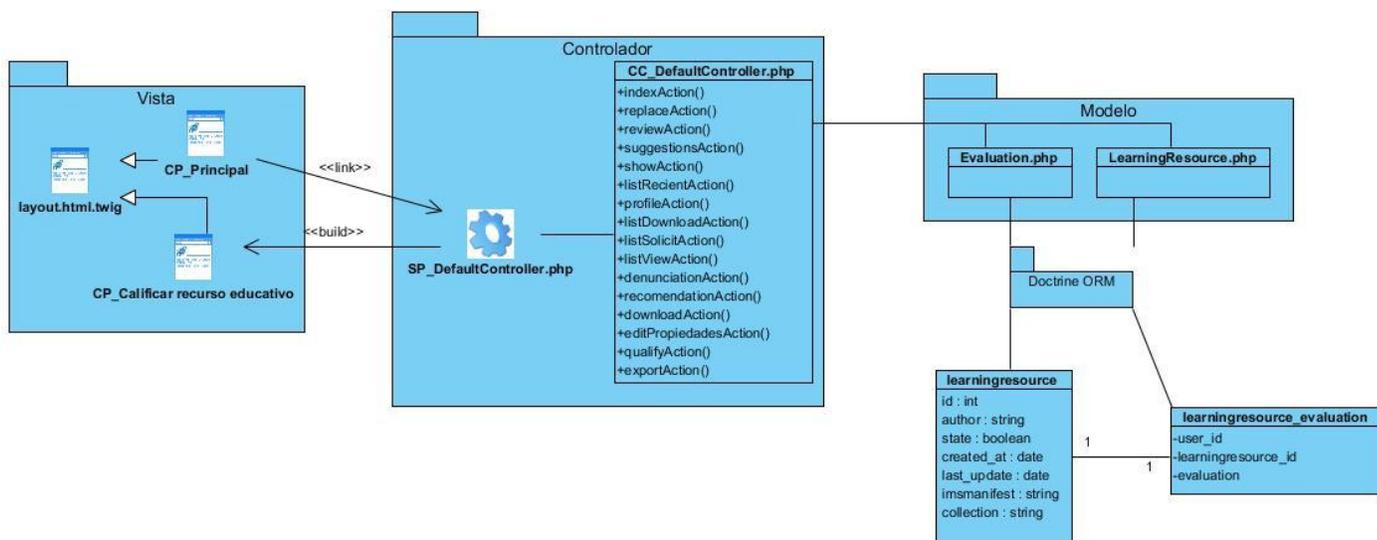
DCD_11: Reemplazar recurso educativo.



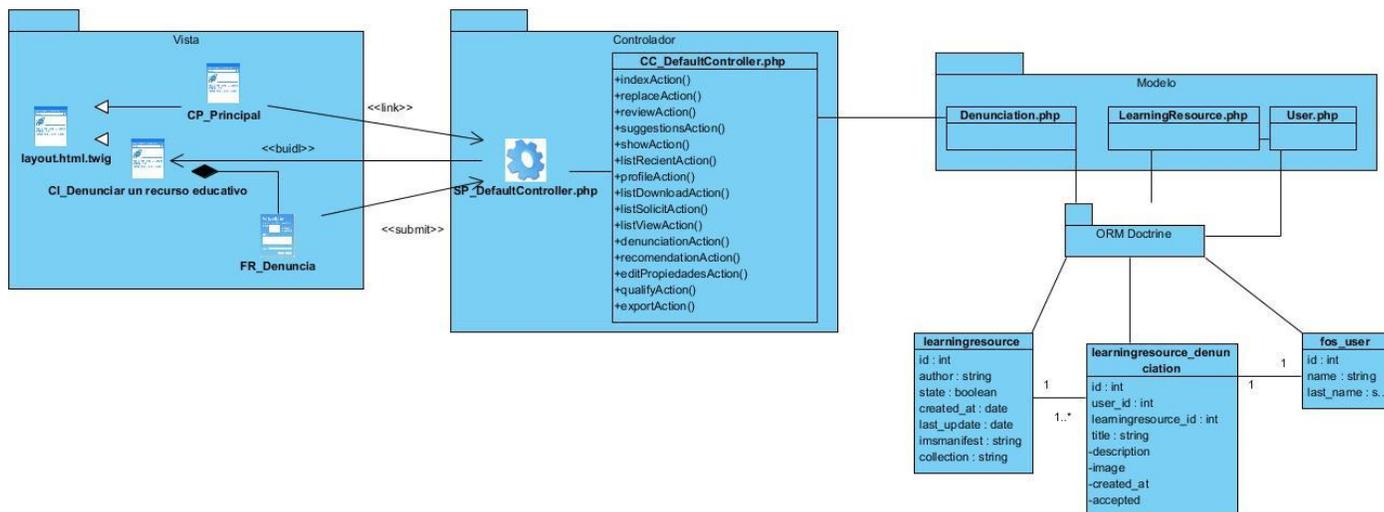
DCD_13: Denunciar comentario de un recurso educativo.



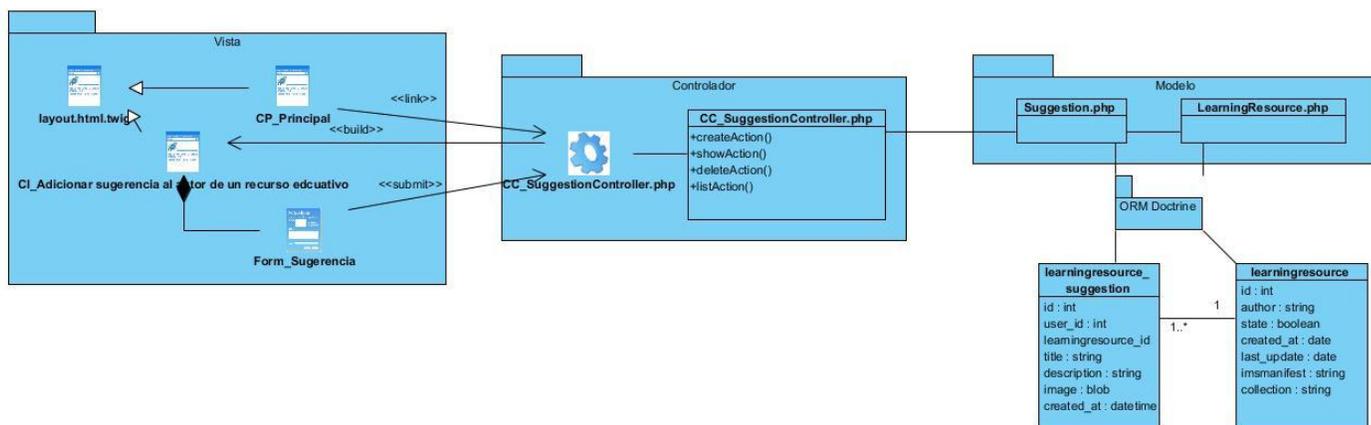
DCD_14: Calificar un recurso educativo.



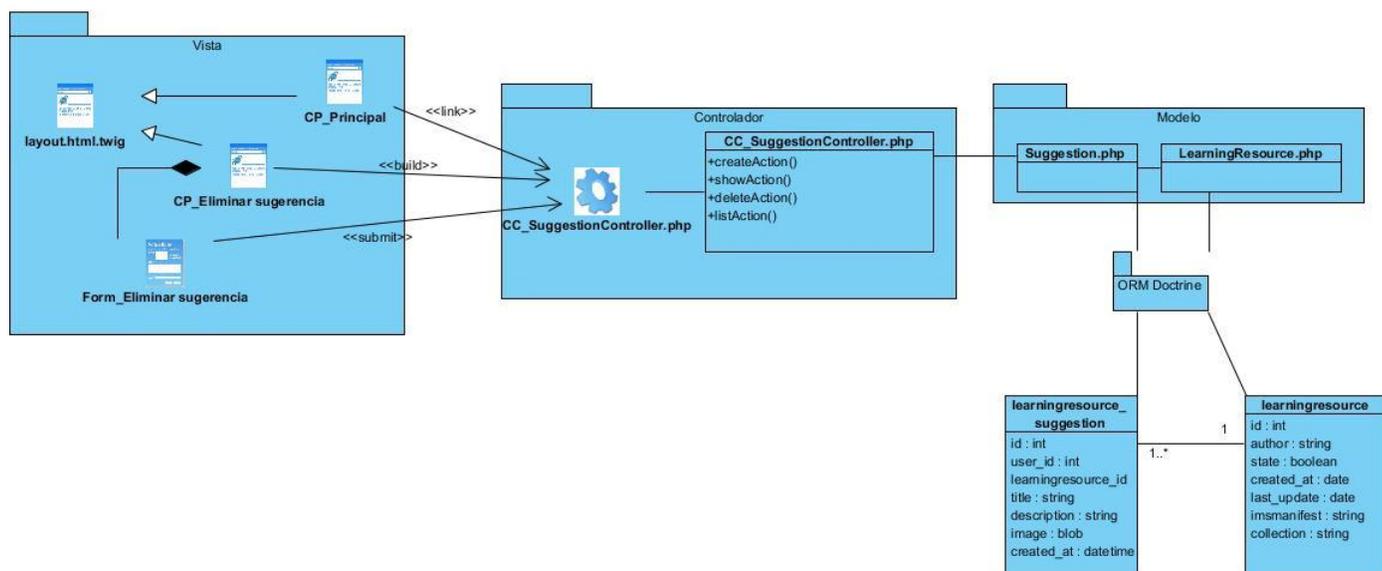
DCD_15: Denunciar un recurso educativo.



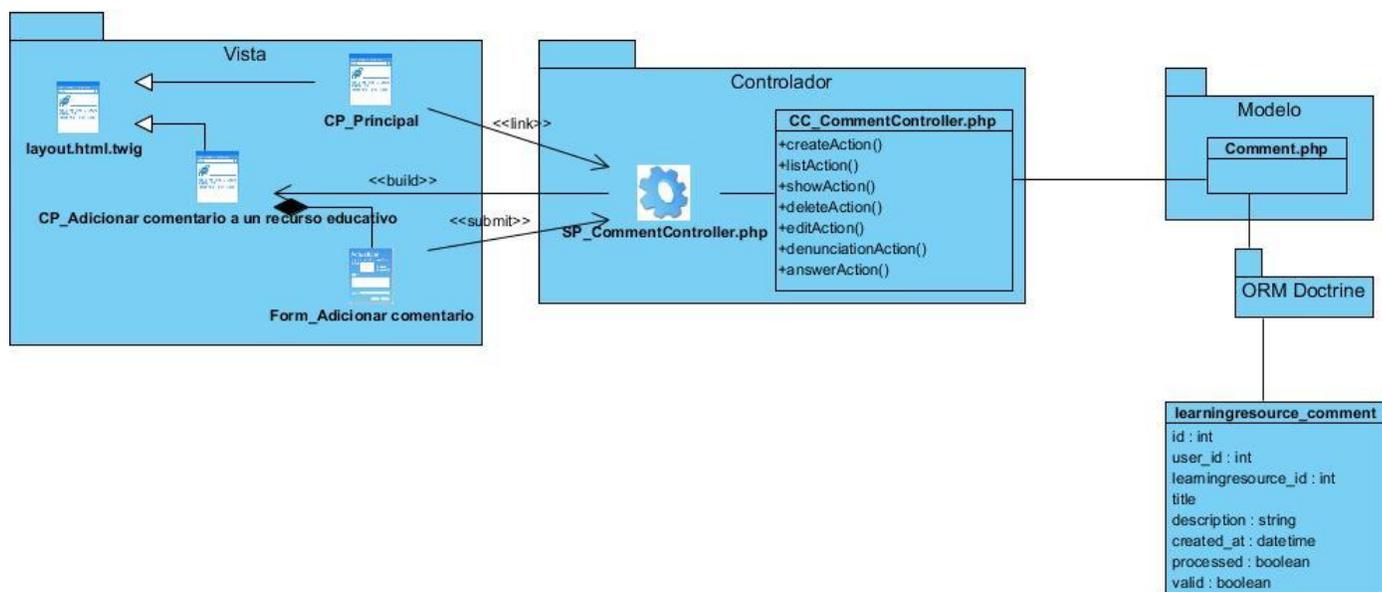
DCD_17: Adicionar sugerencias al autor de un recurso educativo.



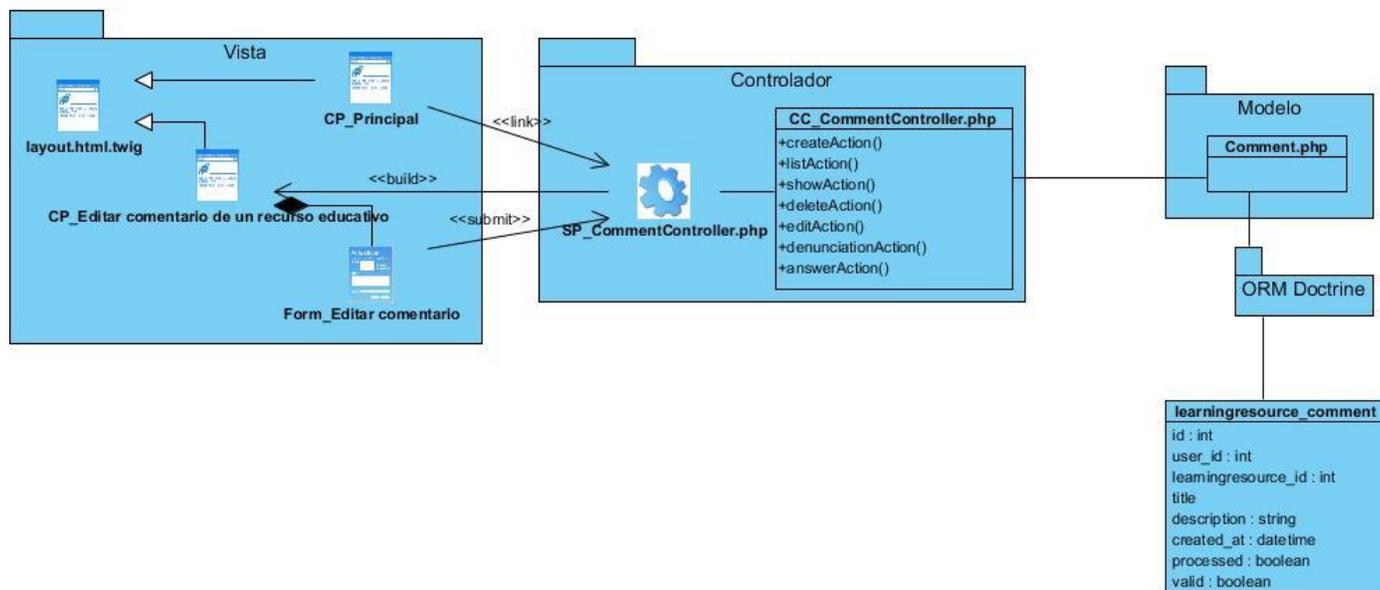
DCD_18: Eliminar sugerencias al autor de un recurso educativo.



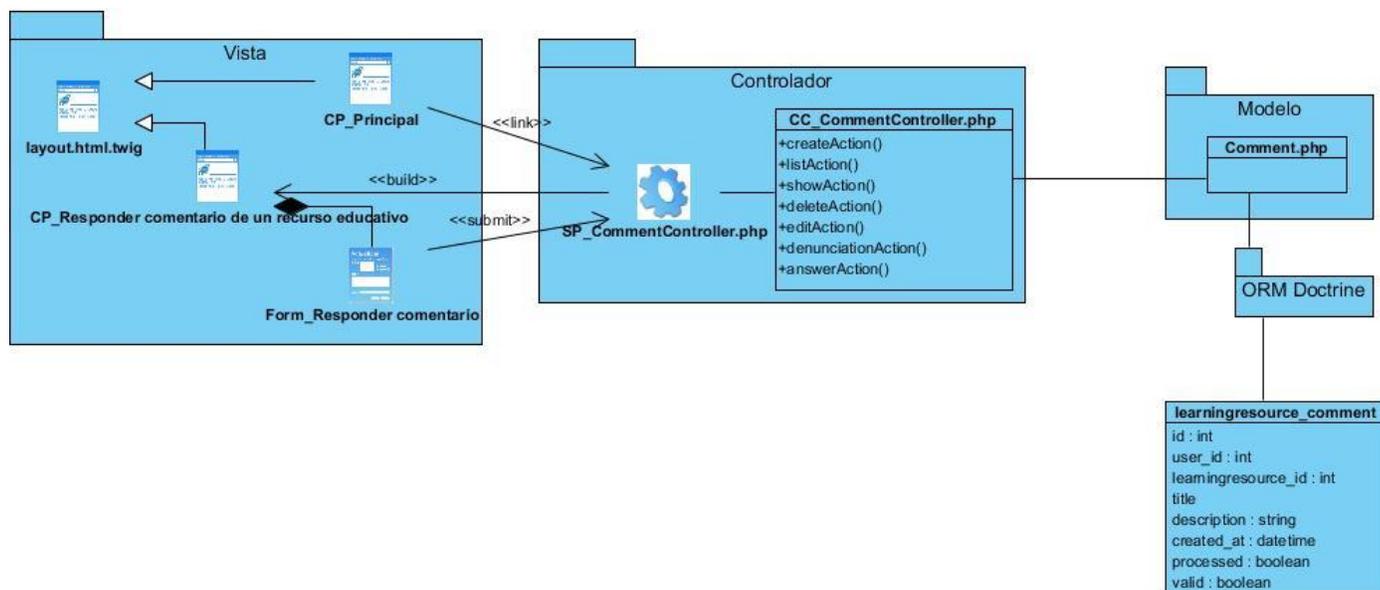
DCD_19: Adicionar comentario a un recurso educativo.



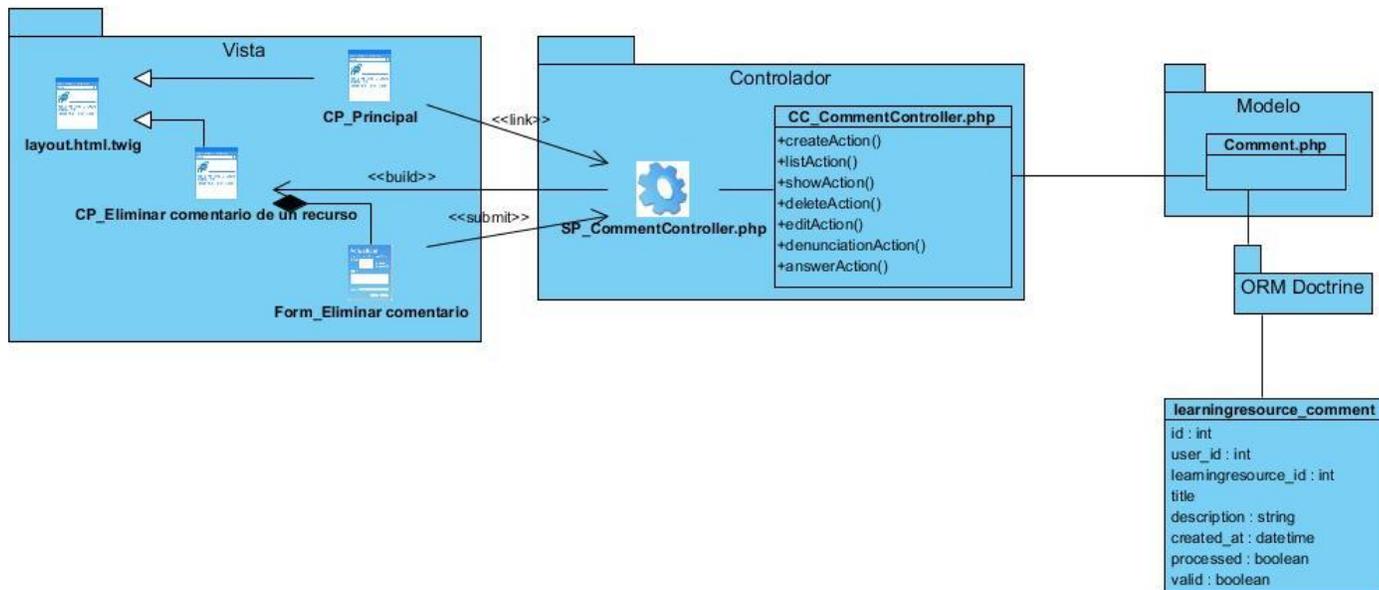
DCD_20: Editar comentario.



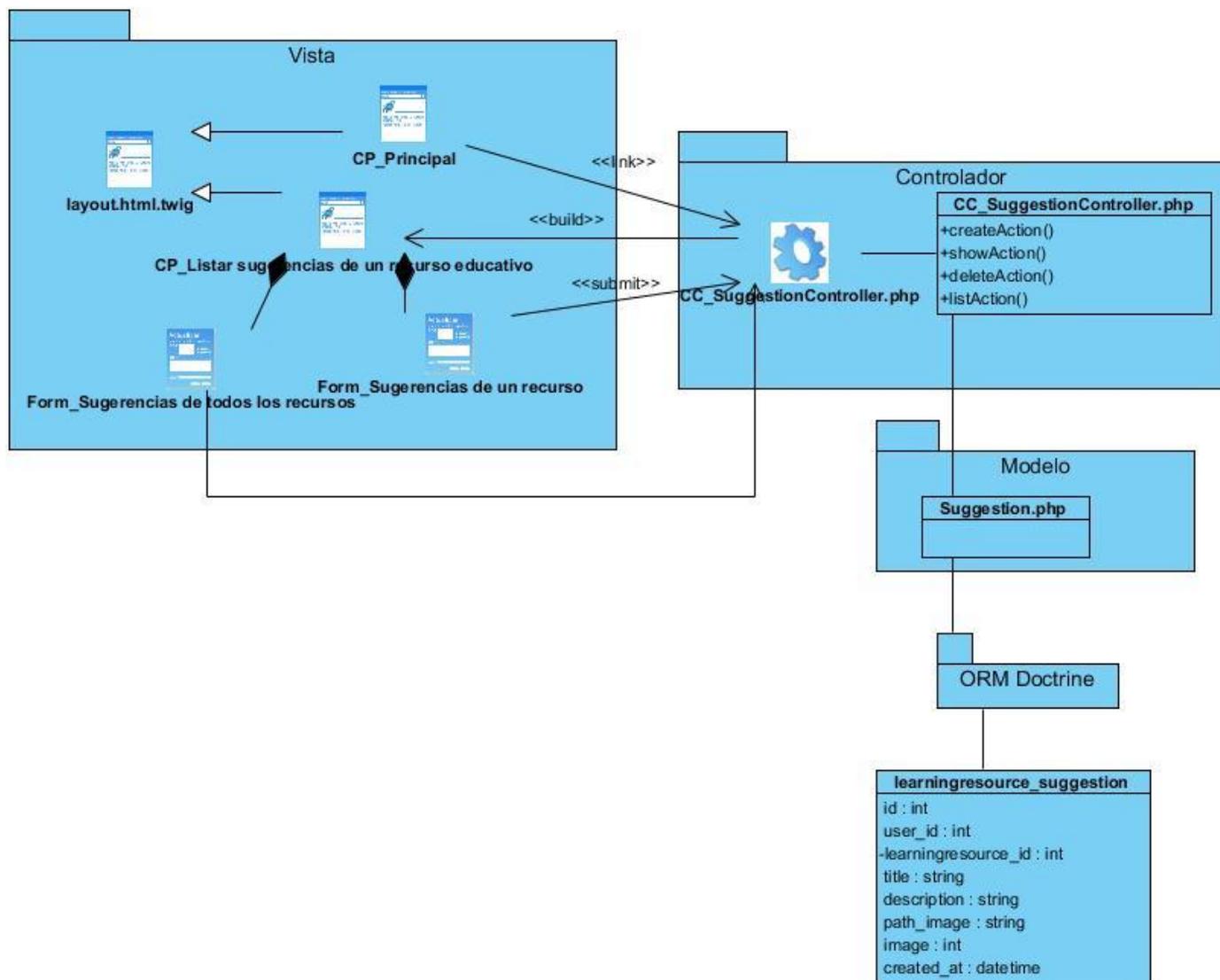
DCD_21: Responder comentario de un recurso educativo.



DCD_22: Eliminar comentario de un recurso educativo.



DCD_23: Listar sugerencia de un recurso educativo.



DCD_24: Adicionar recurso educativo a favoritos.

