

Universidad de las Ciencias Informáticas

Facultad # 4



Implementación del módulo Diseño de aprendizaje para CRODA 3.0.



Autor: Carlos Arturo Rodón Quintas

Tutores: Ing. Osvaldo Ernesto Stable Vilches

Ing. Yandris Mata Cabrera

La Habana, junio de 2015

“Año 57 de la Revolución”

Declaración de autoría

Declaro ser el autor del presente trabajo de diploma y otorgo a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Firma del autor

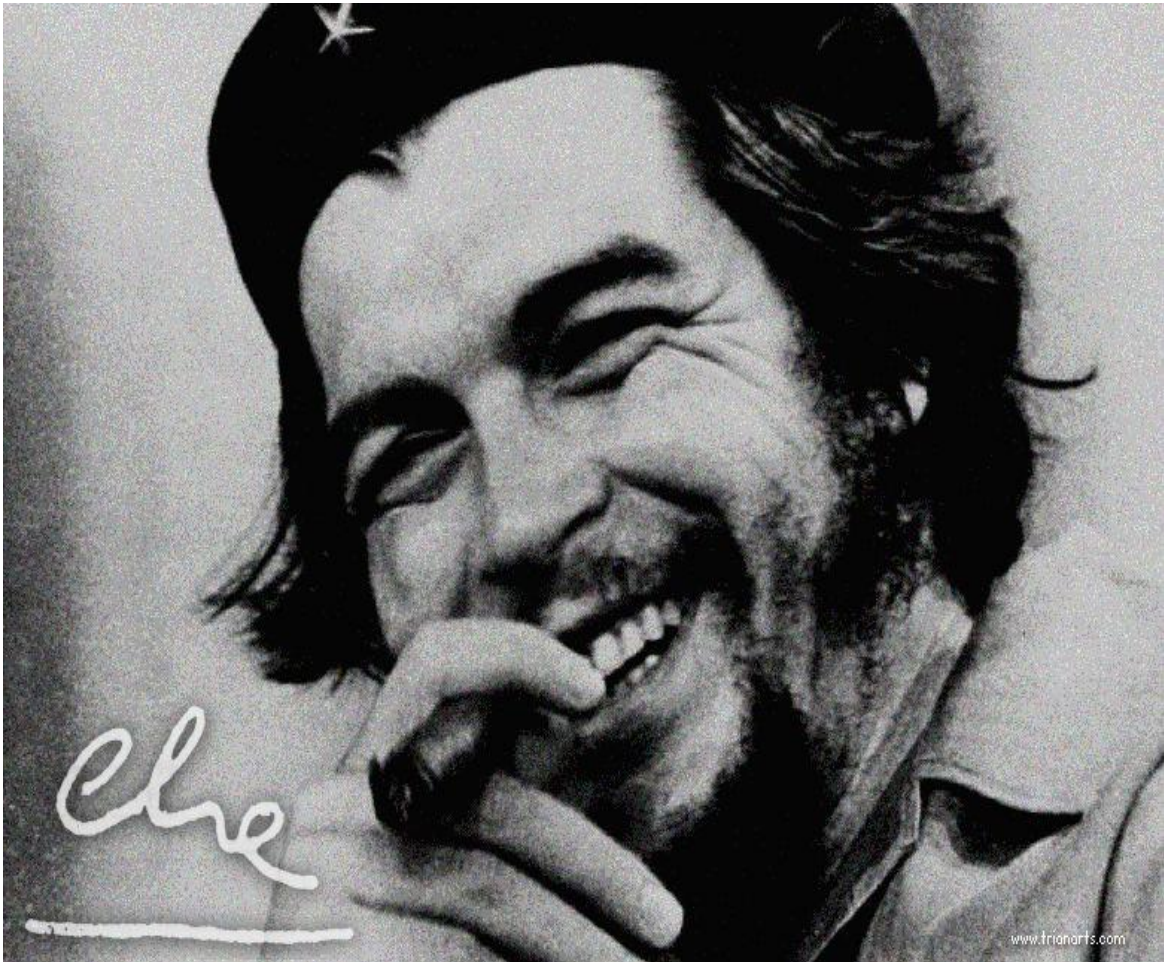
Carlos Arturo Rodón Quintas

Firma del tutor

Ing. Osvaldo Ernesto Stable Vilches

Firma del tutor

Ing. Yandris Mata Cabrera



"Ustedes son jóvenes, creadores de la sociedad perfecta, seres humanos destinados a vivir en un mundo nuevo donde habrá desaparecido definitivamente todo lo viejo"

Dedicatoria

La familia no es algo que se escoge, simplemente la recibes al nacer, pero, si me hubiera tocado elegir sin duda escogería la mía, no puedo pensar una de la que me sienta más orgulloso, por eso dedico esta tesis a mi madre, mi padre y a mis dos hermanos. Gracias por haberme guiado, gracias por ser quienes son.

Agradecimientos

A toda mi familia mis tías, primos, en especial a Gabi, Anita y Fonsi que son casi como mis hermanos.

A mis amigos del apartamento Manuel, Ivan, Yasiel, Yordano, Yudel, Elmer y Manolo.

A los que no pudieron llegar con nosotros hoy a esta graduación pero igual se les recuerda con cariño Chito, Javier, Armandito, Maikel, Yadian, Odel y Nelson.

A las amistades que vienen con uno desde la vocacional en especial a Anita, por todo el apoyo que me ha brindado, sobre todo en este tiempo de tesis.

A Leandro que viene ayudándome desde la vocacional siendo como un hermano mayor para mí.

A mis compañeros de aula, sobre todo a Yeli que ha sido una gran amiga.

A Dairon, Anadelys, Angel Luis, Adrian y Wilder por ser de los mejores amigos que he tenido.

A todos aquellos que de una manera u otra me ayudaron en esta etapa tan importante que es la de la tesis y a todas las amistades que he tenido en estos 5 años.

A mi tutor Osvaldo, muchas gracias por tu ayuda mi hermano.

A los buenos profesores que he tenido, que gracias a ellos logré llegar hasta aquí.

Resumen

La incorporación de las Tecnologías de la Información y las Comunicaciones (TIC) al proceso de enseñanza-aprendizaje sirve de apoyo en la formación de estudiantes en varias materias. Dentro de los recursos que contribuyen a ello se encuentran los diseños de aprendizaje, los que pueden ser creados basándose en la especificación *IMS-Learning Design*, la cual es utilizada en la herramienta de autor web CRODA. El presente trabajo tiene planteado como objetivo principal desarrollar un módulo integrable a CRODA 3.0 que permita la creación de diseños de aprendizaje basándose en la especificación IMS-LD. Para lograrlo se analizaron las tecnologías más factibles a utilizar, se realizó un estudio detallado de la especificación IMS-LD y el análisis y diseño del módulo. Como resultado se obtuvo un módulo que contribuirá a la creación de diseños de aprendizaje basándose en la especificación IMS-LD para la herramienta de autor web CRODA 3.0.

Palabras clave: aprendizaje, CRODA, diseño, especificación, IMS, LD, módulo, tecnologías.

Índice

Introducción	1
Capítulo 1: Fundamentación teórica	6
Introducción.....	6
1.1 <i>E-learning</i>	6
1.2 Lenguajes de modelado educativo	7
1.3 Especificación IMS - <i>Learning Design</i> (IMS - LD)	7
1.3.1 Nivel A	9
1.3.2 Nivel B	10
1.3.3 Nivel C	11
1.4 Diseño de aprendizaje (<i>Learning Design</i>).....	12
1.5 Herramientas de autor.....	14
1.5.1 Reload	14
1.5.2 CopperAuthor	15
1.5.3 CRODA: Herramienta de autor web.....	16
1.6 Herramientas y tecnologías a utilizar	16
1.6.1 Metodología de desarrollo.....	17
1.6.2 Lenguajes de programación y etiquetado.....	18
1.6.3 Herramienta CASE	20
1.6.4 Servidor Web	21
1.6.5 Sistema gestor de bases de datos	21
1.6.6 Entorno de desarrollo integrado (IDE).....	22
1.6.7 <i>Framework</i> de desarrollo Symfony 2.....	23
1.6.8 <i>Framework</i> JavaScript JQuery 1.9.2.....	23
1.6.9 <i>Framework</i> CSS Bootstrap 2.3.....	24
1.7 Conclusiones parciales.....	24
Capítulo 2: Análisis y diseño	25
Introducción.....	25

Índice

2.1 Descripción de la propuesta solución	25
2.2 Requisitos del sistema.....	25
2.2.1 Requisitos funcionales	25
2.2.2 Requisitos no funcionales	27
2.3 Descripción de requisitos	28
2.4 Especificación de requisitos	39
2.5 Modelo de diseño	45
2.5.1 Diagramas de clases del diseño	46
2.6 Patrones de diseño	48
2.7 Patrones Arquitectónicos.....	50
2.8 Modelo de datos.....	53
2.9 Conclusiones parciales.....	53
Capítulo 3: Implementación y pruebas.....	55
Introducción.....	55
3.1 Diagrama de componentes.....	55
3.2 Implementación del sistema	56
3.2.1 Estándares de codificación	56
3.2.2 Principales métodos implementados.....	57
3.3 Validación del módulo Diseño de aprendizaje	60
3.3.1 Pruebas de caja negra	60
3.4 Conclusiones parciales.....	63
Conclusiones generales.....	64
Recomendaciones	65
Referencias bibliográficas	66

Índice de figuras

Figura 1: Estructura de alto nivel de una unidad de aprendizaje en IMS-LD	12
Figura 2: Estructura de un paquete de contenido IMS	13
Figura 3: La estructura de una unidad de aprendizaje en IMS Learning Design.....	14
Figura 4: Prototipo interfaz gráfica de usuario.....	41
Figura 5: Prototipo interfaz gráfica de usuario.....	42
Figura 6: Prototipo interfaz gráfica de usuario.....	45
Figura 7: Diagrama de Clase de Diseño de aprendizaje	47
Figura 8: Clase Entidad LearningDesign.....	47
Figura 9: Clase Entidad Recurso	47
Figura 10: Clase LearningDesignType.....	47
Figura 11: Clase Controladora LearningDesignController	48
Figura 12: Clase controladora DefaultController	48
Figura 13: Funcionamiento del patrón Modelo Vista Controlador.....	52
Figura 14: Patrón Modelo-Vista-Controlador en la propuesta solución.....	52
Figura 15: Modelo físico de la base de datos.....	53
Figura 16: Diagrama de componentes módulo Diseño de aprendizaje	55
Figura 17: Indentación del código	56
Figura 18: Comentarios del código.	57
Figura 19: Convenciones de nombre aplicadas en un método.....	57
Figura 20: Método insertar acto	58
Figura 21: Método guardar_estructuraAction ().....	59
Figura 22: Método cargar_htmlAction()	60

Índice de tablas

Tabla 1: Descripción de los requisitos.....	29
Tabla 2: Especificación de requisito Crear diseño de aprendizaje.....	39
Tabla 3: Especificación de requisito Crear acto	41
Tabla 4: Especificación de requisito Editar acto	43
Tabla 5: CPA Adicionar Diseño de aprendizaje.....	61
Tabla 6: CPA Crear acto del diseño de aprendizaje.....	61
Tabla 7: CPA Editar acto del diseño de aprendizaje	62

Introducción

El desarrollo de la ciencia y la tecnología en la actualidad ha provocado cambios en diversas esferas, dentro de ellas la relacionada al proceso de enseñanza-aprendizaje. Los cambios en esta área están definidos en la forma en que los estudiantes reciben los contenidos que el profesor elabora a través de diferentes medios, tanto clases como cursos. Esta evolución en el escenario educativo hace que la mayoría de las instituciones educacionales se esfuercen por introducir el uso de las Tecnologías de la Información y las Comunicaciones (TIC) en la enseñanza y el aprendizaje, para adaptarse a una nueva demanda social y como motor impulsor para la calidad en la educación.

Gracias a lo anteriormente expuesto se han incrementado las posibilidades de desarrollo de modalidades educativas que permiten, en el ámbito educativo, que surjan propuestas o modelos como materiales didácticos, los cuales apoyan el proceso de enseñanza-aprendizaje. Este elemento, junto al avance de las TIC, demanda cambios en las metodologías y estrategias educativas, para responder con ello a las necesidades de conocimiento de la sociedad.

Con la vinculación de las tecnologías y el proceso de enseñanza-aprendizaje ha surgido el denominado *e-learning* o aprendizaje electrónico, el cual incluye un conjunto de estándares tecnológicos de Internet, métodos y recursos educativos que posibilitan nuevas formas de educación a distancia. El *e-learning* presenta más disponibilidad para los estudiantes y profesores que los métodos tradicionales de enseñanza apoyando en gran medida el proceso de enseñanza-aprendizaje, ya que gracias a las TIC, los estudiantes pueden comunicarse e interactuar con sus compañeros de clase y profesores de una manera más sencilla. Esto les brinda comodidades en sus horarios de estudio proporcionando de esta forma mayor participación del estudiante en las actividades propuestas.

Con el surgimiento del *e-learning* se incrementan las oportunidades de crear y distribuir recursos educativos y que estos a la vez logren ser interpretados por plataformas como: repositorios, sistemas de gestión del aprendizaje y herramientas de autor. Para darle solución a esto se hizo necesaria la creación de una serie de especificaciones y estándares.

En 1997 la Universidad Abierta de los Países Bajos (*Open University of the Netherlands*, OUNL) decidió convertir todos sus cursos en cursos *online*. Los cursos existentes

Introducción

empleaban una variedad de enfoques pedagógicos, y por tanto la Universidad los clasificó y comenzó a implementar unas plantillas representativas que podían dar soporte a todos estos enfoques. Pero rápidamente se hizo evidente que todos los profesores tenían su propia visión pedagógica, y que se necesitaban casi tantas plantillas como profesores. Sin embargo, se observó que aunque había muchas descripciones pedagógicas de los cursos, en la práctica todas consistían en combinaciones de tres elementos básicos: recursos educativos, múltiples personas actuando en varios roles, y actividades pedagógicas. (Griffiths, 2005)

IMS Global, consciente de las limitaciones pedagógicas de las especificaciones existentes en ese momento, comenzó el proceso de desarrollo de una especificación para la definición de aspectos pedagógicos, el resultado fue la creación de una nueva especificación, *IMS-Learning Design*. La cual es un lenguaje de modelado educativo que tiene como objetivo definir formalmente una estructura semántica para anotar los procesos de enseñanza-aprendizaje, y así convertirlos en entidades reutilizables entre diferentes cursos y aplicaciones. (Berlanga, 2005)

Uno de los términos manejados por la especificación IMS-LD es el Diseño de Aprendizaje (*Learning Design*), el cual es definido como “una descripción de un método que permite a los alumnos alcanzar ciertos objetivos de aprendizaje por medio del desarrollo de ciertas actividades de aprendizaje en un cierto orden en el contexto de un cierto ambiente de aprendizaje”. (IMS, 2015)

En el Departamento de Desarrollo de Aplicaciones perteneciente al Centro de Tecnologías para la Formación (FORTES) de la Universidad de las Ciencias Informáticas (UCI), se desarrolla la herramienta de autor web CRODA, actualmente en su versión 3.0, la cual permite la creación de objetos de aprendizaje (OA) reutilizables, accesibles, duraderos e interoperables, la misma utiliza la especificación IMS-LD para la creación de diseños de aprendizaje interoperables y reutilizables. Actualmente CRODA 3.0 no cuenta con un módulo de Diseño de aprendizaje, que permita la creación y edición de estos basándose en la especificación IMS-LD. En la versión anterior de CRODA, CRODA 2.0, existía un módulo para la creación de diseños de aprendizaje, pero este no puede ser incorporado a la actual versión ya que no está desarrollado con tecnología compatible con la utilizada en esta, ya que CRODA 2.0 está desarrollado sobre el *framework* Symfony 1.4 y CRODA 3.0 sobre Symfony 2.3, por lo que no cuenta con ventajas que le permitan integrar este módulo de forma directa a la nueva versión de la herramienta. Además

entrevistas realizadas a personas que han trabajado con la versión anterior del módulo han arrojado como resultado que este editor no era lo suficientemente intuitivo en su utilización. Esto, unido a que su interfaz era complicada provocaba que la curva de aprendizaje fuera demasiado empinada, haciendo difícil lograr una correcta comprensión del módulo y explotación de su potencial al máximo.

Debido a lo anteriormente planteado surge el siguiente **problema a resolver**: ¿Cómo contribuir a la edición de diseños de aprendizaje basados en la especificación IMS-LD en la herramienta de autor web CRODA 3.0?

Se define como **objeto de estudio** de la presente investigación el proceso de creación de diseños de aprendizaje basándose en la especificación IMS-LD.

Como **campo de acción**, la creación de diseños de aprendizaje en CRODA 3.0 basándose en la especificación IMS-LD.

El **objetivo general** de este trabajo es desarrollar un componente integrable para CRODA 3.0 que permita la creación de diseños de aprendizaje basándose en la especificación IMS-LD.

El cual es posible desglosar en los siguientes **objetivos específicos**:

- Elaborar el marco teórico conceptual relacionado con la creación de diseños de aprendizaje basados en la especificación IMS-LD.
- Estudiar las principales características de los diseños de aprendizaje basados en la especificación IMS-LD y las principales funcionalidades, ventajas y deficiencias presentes en el módulo Diseño de aprendizaje de CRODA 2.0
- Desarrollar el módulo Diseño de aprendizaje integrable a CRODA 3.0.
- Validar el correcto funcionamiento de las funcionalidades de la propuesta solución e integrar el módulo Diseño de aprendizaje a CRODA 3.0.

Para guiar el desarrollo de este trabajo se plantean las siguientes **preguntas científicas**:

- ¿Cuáles son las tecnologías necesarias para el desarrollo de un módulo para la creación de diseños de aprendizaje basados en IMS-LD para CRODA 3.0?
- ¿Cómo determinar las funcionalidades que debe brindar un módulo para la creación de diseños de aprendizaje basados en IMS-LD para CRODA 3.0?
- ¿Cómo crear los componentes y funcionalidades en el módulo para la creación de diseños de aprendizaje basados en IMS-LD para CRODA 3.0?
- ¿Es recomendable la propuesta de solución desarrollada?

Con el objetivo de proveer las bases necesarias de la investigación se hizo necesaria la utilización de los siguientes **métodos de investigación**.

Métodos teóricos:

- **Analítico–sintético:** permitió el estudio y análisis de los diferentes conceptos y definiciones permitiendo la extracción de los elementos más importantes relacionados con el proceso de creación de diseños de aprendizaje basados en la especificación IMS-LD.
- **Análisis Histórico–Lógico:** permitió constatar teóricamente cómo ha evolucionado el proceso de creación de los diseños de aprendizaje basados en IMS-LD en diferentes editores, sus características y ventajas, para evaluar posibles mejoras a incluir en la aplicación.

Métodos empíricos:

- **Observación:** se utilizó para obtener información de las necesidades existentes a lo largo del desarrollo del proyecto.
- **Entrevista:** realizada a usuarios de la anterior versión del módulo Diseño de aprendizaje, para detectar las principales características, ventajas y deficiencias del mismo.

Se tiene como **posible resultado** obtener un componente integrable a CRODA 3.0 para la creación de diseños de aprendizaje.

La investigación se desglosa en tres capítulos, en los cuales se abordan de forma simplificada los siguientes contenidos:

Capítulo I Fundamentación teórica de la investigación: se abordan de forma general los principales conceptos a tener en cuenta en la investigación. Se hace un estudio de la especificación IMS–LD y el análisis de la metodología y herramientas utilizadas para el desarrollo y diseño del módulo con el objetivo de proponer el más indicado para su utilización.

Capítulo II Análisis y Diseño del módulo: contiene el levantamiento de requisitos, obteniendo los principales artefactos que genera los flujos de trabajo propuestos por la metodología a utilizar. Se realiza el análisis y diseño del módulo, así como los diagramas de clases del diseño, que brindan una visión clara de cómo debe quedar el producto luego

Introducción

de realizada la investigación. Se muestran los patrones arquitectónicos y de diseño utilizados.

Capítulo III Implementación y pruebas del módulo: se describe cómo está implementado el módulo, donde se observa de forma general el cumplimiento de los objetivos definidos anteriormente. Se describen además las pruebas realizadas al módulo para comprobar su correcto funcionamiento.

Capítulo 1: Fundamentación teórica

Capítulo 1: Fundamentación teórica

Introducción

En el presente capítulo se relacionan los conceptos necesarios para poder concebir el módulo de creación de diseños de aprendizaje de la herramienta de autor CRODA 3.0, basado en la especificación IMS-LD. Se realiza un estudio de las diferentes herramientas de autor que basan su funcionamiento en IMS-LD. Además, se presenta una descripción de las diferentes tecnologías y herramientas necesarias, con el objetivo de proponer las más indicadas para el desarrollo del módulo.

1.1 E-learning

El *e-learning* es definido por (e-ABC, 2010) como “el *e-learning* consiste en la educación y capacitación a través de Internet. Este tipo de enseñanza *online* permite la interacción del usuario con el material mediante la utilización de diversas herramientas informáticas.”

El Centro de formación permanente de la Universidad de Sevilla lo define como “procesos de enseñanza-aprendizaje que se llevan a cabo a través de Internet, caracterizados por una separación física entre profesorado y estudiantes, pero con el predominio de una comunicación tanto síncrona como asíncrona, a través de la cual se lleva a cabo una interacción didáctica continuada. Además, el alumno pasa a ser el centro de la formación, al tener que autogestionar su aprendizaje, con ayuda de tutores y compañeros.” (CFP, 2015) Esta definición será la seleccionada por el autor para utilizarla durante el transcurso de la investigación al considerarla más completa.

El *e-learning* emplea herramientas o aplicaciones de hipertexto como por ejemplo: correo electrónico, páginas web, foros de discusión, chat, plataformas de formación, como soporte de los procesos de enseñanza-aprendizaje, aportándoles variados beneficios en la superación de los profesores debido a las facilidades que brinda de poder recibir cursos, clases y conferencias a través de la red.

Brinda beneficios como la reducción de costos al permitir disminuir y eliminar gastos a través de la red, se caracteriza por su rapidez y agilidad en las comunicaciones de los sistemas en la red, su contenido es accesible desde cualquier conexión a Internet cuando el usuario lo necesite, es flexible ya que no requiere que un grupo de personas coincidan

Capítulo 1: Fundamentación teórica

en tiempo y espacio en un lugar determinado para poder intercambiar sus ideas y conocimientos.

1.2 Lenguajes de modelado educativo

La generalización del término EML en *e-learning* proviene del trabajo desarrollado en la Universidad Abierta de los Países Bajos (*Open University of the Netherlands*, OUNL) a finales de los años 90. El grupo de investigación liderado por el Profesor Rob Koper analizó los sistemas de gestión de la enseñanza (LMS de su término en inglés *Learning Management Systems*) que existían y que eran los más utilizados en aquella época, intentando identificar los problemas y defectos de dichos sistemas de *e-learning*. En particular se identificó como principal problema la falta de aplicación de la teoría instruccional y del aprendizaje dentro de los mismos. Como resultado desarrolló y puso en práctica una propuesta basada en la definición de un Lenguaje específico de dominio llamado *Educational Modelling Language* (EML). (Instituto de tecnologías educativas, 2015)

Los EML son lenguajes que permiten describir UOL (*Unit of learning*, en español unidad de aprendizaje), las cuáles a su vez describen el proceso de aprendizaje como un todo (y no solo centrado en los contenidos como se hace en los objetos de aprendizaje). Otra característica añadida de estos lenguajes es que proporcionan un mecanismo para la comunicación entre el personal técnico de soporte y el personal no técnico (normalmente los educadores) dentro de una organización durante la operacionalización del EML. Ahora las UOL son completas de modo que el personal técnico puede saber qué es lo que está planificado y ayudar a resolver las incidencias que pudieran producirse como podría ser la no disponibilidad de un recurso.

1.3 Especificación IMS - *Learning Design* (IMS - LD)

La especificación IMS-LD es un lenguaje de modelado educativo que tiene como objetivo definir formalmente una estructura semántica para anotar los procesos de enseñanza-aprendizaje, para así convertirlos en entidades reutilizables entre diferentes cursos y aplicaciones. Para ello describe procesos que, con la intención de alcanzar un objetivo de aprendizaje, agrupan las actividades que realizarán alumnos y profesores, indicando en qué momento, con qué recursos didácticos y servicios, y bajo qué condiciones lo harán.

Capítulo 1: Fundamentación teórica

IMS-LD define formalmente una estructura para describir cualquier diseño de enseñanza-aprendizaje.

Concretamente, esta especificación pretende describir totalmente el proceso de enseñanza-aprendizaje de una unidad de aprendizaje. Anotar el significado y la funcionalidad pedagógica de los elementos, marcar aspectos de personalización para que las actividades puedan adaptarse a las preferencias, conocimientos previos o necesidades educativas de los usuarios. También describir de manera abstracta el diseño de aprendizaje para poderlo repetir en diferentes condiciones y con diferentes personas, identificar, descontextualizar e intercambiar elementos de aprendizaje y reutilizarlos en otros contextos, fomentando el intercambio y uso de información entre diferentes aplicaciones compatibles con IMS-LD. (Berlanga, 2005)

IMS-LD permite diseñar las experiencias educativas a las que se expondrán a los usuarios de las aplicaciones de *e-learning*. Aun siendo un lenguaje informático artificial, el lenguaje de IMS-LD contiene primitivas y conceptos que pueden ser fácilmente entendidos por los educadores. De esta forma, su uso no demanda conocimientos muy avanzados de informática, programación o desarrollo de aplicaciones web. Por el contrario, con un poco de entrenamiento y con un mínimo de herramientas de soporte (por ejemplo, editores específicos para IMS-LD), un educador motivado podrá utilizar el lenguaje para plasmar sus diseños educativos. Estos diseños servirán como base para la generación automática de las aplicaciones finales utilizando herramientas informáticas adecuadas. De esta forma, el principal potencial de IMS-LD es permitir que sean los propios educadores, que son los que realmente entienden las necesidades y objetivos finales de las aplicaciones educativas, los que lideren el desarrollo y mantenimiento de dichas aplicaciones. (Instituto de tecnologías educativas, 2015)

Para facilitar su adopción progresiva, la especificación propone tres niveles de detalle a los que denomina simplemente 1, 2 y 3 o A, B y C. De este modo, el primer nivel es bastante sencillo de implementar y permite crear diseños de aprendizaje sencillos. Un LMS que implemente solamente el nivel A no puede considerarse completamente compatible con la especificación IMS-LD, pero sí puede considerarse compatible con el Nivel A de la especificación. Los Niveles B y C añaden funcionalidad y potencia, construyendo siempre sobre el nivel anterior. A continuación se explican con mayor profundidad estos 3 niveles.

Capítulo 1: Fundamentación teórica

1.3.1 Nivel A

El Nivel A de la especificación se centra en superar el modelo de un único usuario reflejado en el resto de las especificaciones de IMS. En este primer nivel de la especificación se incluyen los conceptos básicos expuestos en la sección anterior, esto es, las obras, divididas en actos en las que distintos actores interpretan distintos roles.

La noción de estructuras de actividades, que es la esencia de la definición de los caminos de aprendizaje con ramificaciones, también aparece en el Nivel A. Con esta información es posible crear unidades de aprendizaje en las que se define un proceso colaborativo en el que participan varios actores, como estudiantes y profesores, y se define un secuenciamiento complejo de las actividades en el que en algunos casos importa el orden y en otros no. Lo que no se incluye en el Nivel A es la posibilidad de modificar y consultar valores, con lo que los flujos de aprendizaje son fijos y el resultado de las distintas actividades no puede afectar al resto.

Aun así, una implementación que solo soporte el Nivel A podría tolerar un modelo en el que aparezcan distintos tipos de participantes que realizan diversas actividades en un determinado orden. Por tanto, este nivel ya presenta un aporte sobre el modelo dirigido a un único tipo de usuario y abre la puerta a diseños de aprendizaje basados en los principios del aprendizaje colaborativo.

Por otro lado, dado que otra posible interpretación de los roles es la de distintos perfiles de alumnos, el Nivel A de IMS-LD soporta modelos educativos en los que distintos tipos de alumno recorren diferentes caminos al realizar un determinado curso.

El nivel A está conformado por los siguientes elementos (Falcon, 2011):

- Persona: es la persona que adopta un determinado rol.
- Rol: por defecto son, aprendiz y grupo de trabajo.
- Actividad: existen dos tipos de actividades, de apoyo y de aprendizaje, así como estructuras de actividades que las agrupan.
- Ambiente: es una colección estructurada de servicios, objetos de aprendizaje y sub-ambientes.

Capítulo 1: Fundamentación teórica

- Método: es el mecanismo que permite relacionar los roles, las actividades y los ambientes. Se compone de tres elementos:
 - Obra: un método se compone de una o más obras, y estas a su vez de uno o varios actos.
 - Acto: es aquí donde se establecen las distintas actividades en Partes del rol adecuadas. Un acto involucra a un participante o a varios, pero cada uno de ellos lleva a cabo una única actividad.
 - Parte del rol: asocia únicamente un solo rol a una única actividad, teniendo como objetivo, especificar lo que ese rol debe cumplir en el Acto.

1.3.2 Nivel B

Los dos aportes fundamentales del Nivel B de la especificación IMS-LD son las propiedades y las condiciones.

Las propiedades son pares atributo-valor que parten de un estado inicial y se modifican a lo largo del proceso de ejecución de la unidad de aprendizaje. Un ejemplo sería examen-superado con un valor inicial de falso. Durante la actividad de evaluación del examen es posible que este valor se convierta en verdadero o que se quede en su estado inicial.

En cuanto a las condiciones, éstas son consultas que se realizan sobre el valor de las propiedades en un momento determinado. Así, para llegar al estado final del ejemplo de la sección anterior, es necesario que la propiedad examen-superado tome el valor verdadero. Si tras la actividad de evaluación del examen el valor siguiese siendo falso, el alumno deberá recorrer el camino de aprendizaje nuevamente.

Así, el Nivel B aporta la posibilidad de que el resultado de una actividad genere un cambio en alguna de las propiedades. Por su parte, el resto de las actividades pueden estar condicionadas a un cierto valor de las propiedades. En la práctica esto significa que el resultado de unas actividades puede tener un impacto real en el resto del proceso de aprendizaje, cambiando el camino a seguir o incluso modificando el propio contenido de alguna actividad. (Instituto de tecnologías educativas, 2015)

IMS-LD define cuatro tipos de propiedades:

Capítulo 1: Fundamentación teórica

- **Propiedades locales:** se almacenan con un alcance local para la ejecución de una unidad de aprendizaje. Se definen y utilizan en la unidad de aprendizaje. El valor de esta propiedad es el mismo para todos los usuarios en el funcionamiento de la unidad de aprendizaje, pero pueden variar en diferentes ejecuciones.
- **Propiedades globales:** son accesibles fuera del contexto de una unidad de aprendizaje, es decir en más de una unidad de aprendizaje. Se pueden definir en una unidad de aprendizaje y utilizarse en otra. Las propiedades globales una vez definidas no pueden ser cambiadas.
- **Propiedades personales:** son propiedad de una persona (local o global). Estas propiedades se utilizan para la personalización. Por ejemplo, una cartera que funciona a través de unidades de aprendizaje puede ser modelado con propiedades globales personales. Las propiedades personales pueden ser almacenadas en un expediente portátil personal.
- **Propiedades de roles:** son propiedad de una función y son siempre locales. Todos los usuarios en un rol específico pueden acceder a esta propiedad y tiene el mismo valor en el mismo plazo de la unidad de aprendizaje.

1.3.3 Nivel C

La adición de propiedades y condiciones en el Nivel B de la especificación permite la creación de unidades de aprendizaje cuyo recorrido cambia durante la propia ejecución; pero estas actividades se ejecutan en un determinado orden y esperan a que la actividad anterior termine antes de comenzar su ejecución.

El Nivel C de la especificación introduce un mecanismo de notificación o envío de mensajes entre las distintas actividades. Esto significa que una actividad puede estar ejecutándose en determinadas condiciones y en un momento no predecible recibir un mensaje desde otra actividad o desde el propio LMS que afecte la ejecución de la actividad inicial.

Esto permite soportar flujos de aprendizaje modificables en tiempo real mediante eventos. Los flujos pre-definidos se sustituyen por actividades que se disparan, modifican o interrumpen a medida que cambia el estado de la unidad de aprendizaje.

Capítulo 1: Fundamentación teórica

Dado que en estos procesos de aprendizaje normalmente hay varios individuos, el camino que se seguirá y el orden de ejecución de las actividades ya no es predecible, pues es alterado por la acción de los distintos roles. (Instituto de tecnologías educativas, 2015)

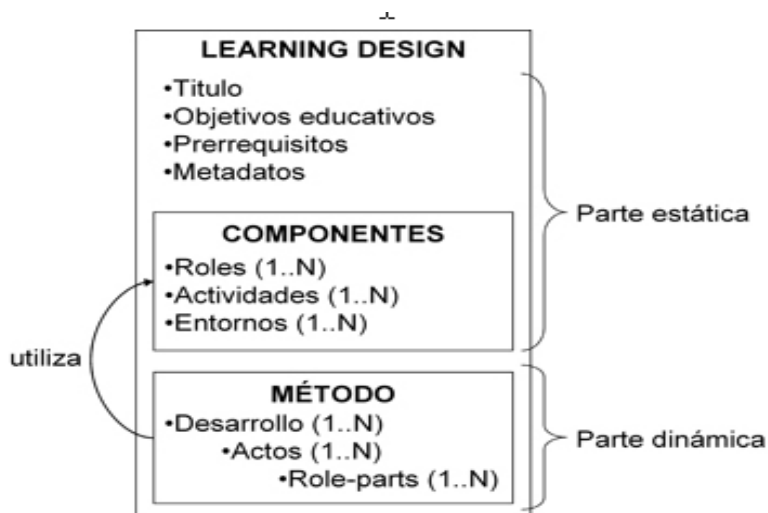


Figura 1: Estructura de alto nivel de una unidad de aprendizaje en IMS-LD

1.4 Diseño de aprendizaje (*Learning Design*)

Un diseño de aprendizaje es definido como “una descripción de un método que permite a los alumnos alcanzar ciertos objetivos de aprendizaje por medio del desarrollo de ciertas actividades de aprendizaje en un cierto orden en el contexto de un cierto ambiente de aprendizaje”. (IMS, 2015)

El término diseño de aprendizaje se refiere al conocimiento que emplean los profesores o diseñadores cuando definen y diseñan instrucción. Tradicionalmente, este conocimiento se ha aplicado y difundido mediante modelos que prescriben cómo diseñar la enseñanza para desarrollar las habilidades cognitivas, emocionales y físicas de los estudiante. (Berlanga, 2005)

Los diseños de aprendizaje se definen empleando el lenguaje formalizado en la especificación IMS-LD incluyendo únicamente el proceso de enseñanza-aprendizaje, pero no define ni contiene los materiales o recursos que dicho proceso utiliza. El diseño de un curso en sí no es un recurso con el que se pueda aprender, pues las actividades a menudo requieren contenido que debe ser distribuido junto al diseño. Dentro de la familia

Capítulo 1: Fundamentación teórica

de especificaciones de IMS, se propone que los diseños de aprendizaje se distribuyan junto con sus contenidos asociados en forma de paquete siguiendo la especificación *IMS-Content Packaging*. A estos paquetes que unen diseño y contenido se les denomina unidades de aprendizaje (UOL). (Instituto de tecnologías educativas, 2015)

Una unidad de aprendizaje según (Koper and Tattersall, 2005) es: “Una unidad de educación o formación completa y autónoma, curso, módulo o lección. La creación de una unidad de aprendizaje implica la creación de un diseño de aprendizaje y también la compilación de sus recursos asociados, bien como archivos contenidos en la unidad o como referencias web, incluyendo evaluaciones, materiales de aprendizaje e información para configurar el servicio de aprendizaje”.

En sí, una UOL contiene todos los recursos asociados como ejercicios, objetos o recursos de aprendizaje e información para configurar distintos servicios (ej. conferencia, correo electrónico, búsqueda). Un escenario de aprendizaje completo viene definido por los recursos, la metodología, las actividades de aprendizaje y soporte, los servicios complementarios, la relación entre los roles, la agrupación de usuarios, entre otros. Todo ello se define en un fichero llamado manifiesto, escrito en XML, que junto con los recursos es empaquetado en un fichero en formato ZIP, consiguiendo la unidad de aprendizaje. Este fichero comprimido podrá ser ejecutado, abierto o modificado en cualquier sistema compatible LD. (Burgos, 2005)



Figura 2: Estructura de un paquete de contenido IMS

Capítulo 1: Fundamentación teórica



Figura 3: La estructura de una unidad de aprendizaje en IMS-Learning Design

1.5 Herramientas de autor

Las herramientas de autor son aplicaciones que permiten la creación de contenidos educativos como cursos o lecciones que sirven de apoyo a profesores y estudiantes en el proceso de enseñanza-aprendizaje, adecuándolo a las necesidades específicas que el autor desee. Estas permiten controlar el aprendizaje, mediante cuestionarios y marcadores que evalúan el conocimiento que el estudiante vaya adquiriendo.

Estas se pueden clasificar en tres tipos, las que permiten la creación de materiales educativos digitales, las que pueden generar todos los materiales a incluir en el curso y su publicación y las que generan simulaciones. (Montero, 2008).

Actualmente existen diversas herramientas de autor compatibles con la especificación IMS-LD como son Reload, CoS-MoS, Alfabet, CopperAuthor, Collage y ReCourse. A continuación se analiza Reload y CooperAuthor ya que las mismas utilizan los 3 niveles de la especificación para la creación de diseños de aprendizaje.

1.5.1 Reload

Reload se enfoca en el desarrollo de herramientas que están basadas en las emergentes especificaciones de interoperabilidad de tecnologías de aprendizaje. Es desarrollado por

Capítulo 1: Fundamentación teórica

las universidades de Bolton y Strathclyde. Los objetivos primarios de este proyecto son facilitar la creación, el compartir y la reutilización de servicios y objetos de aprendizaje, y aumentar el rango de metas pedagógicas realizables a través de la planificación de lecciones.

Esos objetivos se lograrán a través de la producción de un conjunto de programas que servirán como herramientas de autor, así como reproductores de objetos de aprendizaje.

La suite de programas de Reload está compuesta actualmente por:

- Editor Reload Clásico. Este editor soporta IMS-LD, IEEE LOM, paquetes de contenido IMS 1.1.4, Scorm 1.2 y Scorm 2004, existiendo versiones compatibles con Windows, Linux y Mac.
- Editor Reload basado en Eclipse, IMS-LD (versiones 1.1, 1.2, 1.2.4), IEEE LOM, IMS CP (versiones 1.1.1, 1.1.2, 1.1.4), paquetes de contenido IMS 1.1.4, SCORM 1.2, y SCORM 2004 (3ra edición)
- Editor de Diseño de Aprendizaje basado en Eclipse Rich Client Platform e IMS-LD versión 1.0

1.5.2 CopperAuthor

CopperAuthor es una herramienta desarrollada en paralelo al motor de ejecución CopperCore. Esta herramienta permite a los diseñadores construir y navegar sobre la estructura del diseño educativo mediante una interfaz basada en tablas. En su versión actual la interfaz es un tanto primitiva y permite el desarrollo de diseños educativos IMS-LD de los 3 niveles. (ITE, 2010)

La herramienta de autor CopperAuthor define los usuarios, las actividades de aprendizaje, las actividades de soporte y demás elementos correspondientes al nivel A acorde con la especificación. Ofrece funcionalidades para incorporar el servicio de monitorización y elementos globales, lo que permite al usuario definir estructuras más complejas. Brinda flexibilidad a la hora de la representación didáctica, ya que permite esconder y mostrar elementos, condicionar el flujo de aprendizaje, almacenar datos del usuario, tanto a nivel local como personal.

Los creadores de unidades de aprendizaje, que pueden no tener conocimiento de LD, necesitan ser capaces de representar su experiencia pedagógica y los planes de

Capítulo 1: Fundamentación teórica

instrucción mediante el uso de sus vocabularios convencionales como el curso, la clase o la presentación.

Para estos profesores sin conocimientos previos de informática y de la especificación, dichas herramientas poseen una organización compleja y un vocabulario no adecuado.

1.5.3 CRODA: Herramienta de autor web

El principal objetivo de esta herramienta es facilitar la creación de OA empleando el estándar SCORM, garantizando así la interoperabilidad y reusabilidad de los recursos educativos en diferentes aplicaciones que utilicen este propio estándar.

Algunas de sus características y funcionalidades hasta el momento:

- Permite el acceso a diferentes servicios manteniendo una simplicidad en su arquitectura.
- Brinda la posibilidad de crear plantillas: estas son solamente una estructura inicial para la creación de algún OA. Siendo posible además, luego de haber creado dichas plantillas hacerlas públicas, para otras personas que interactúen con el sitio, luego de haber sido analizadas y revisadas por una persona calificada en el tema.
- Cuando se crea algún OA, dicha herramienta brinda la posibilidad de modificar la estructura inicial de las plantillas, permitiendo incluir nueva información y contenido.
- A los OA que son creados, se les incorpora metadatos que describen algunos de sus elementos facilitando así la reusabilidad e identificación en otras herramientas.
- La incorporación de los metadatos se implementa con el uso del estándar LOM. Esta herramienta de autor presenta comunicación con un Repositorio de Objetos de Aprendizaje (ROA), brindándole la posibilidad de guardar los recursos educativos una vez creados. En dicho repositorio se pueden realizar búsquedas con diferentes criterios, con el fin de pasar al área de creación aquel OA, al cual se le desee modificar su contenido.

1.6 Herramientas y tecnologías a utilizar

Para la realización del módulo Diseño de aprendizaje para CRODA 3.0 se hace necesaria la descripción de un conjunto de tecnologías y herramientas, las mismas son definidas y

Capítulo 1: Fundamentación teórica

utilizadas por el proyecto CRODA 3.0 para el cual será desarrollado el módulo Diseño de aprendizaje.

1.6.1 Metodología de desarrollo

En la actualidad existen diversas metodologías para el desarrollo de software, se hace necesaria la aplicación de los procedimientos que proponen las mismas para llevar a cabo un proceso de desarrollo de software bien estructurado, planeado y controlado, y así obtener un producto con una calidad de excelencia.

Se entiende por metodología de desarrollo “una colección de documentación formal referente a los procesos, las políticas y los procedimientos que intervienen en el desarrollo del software. La finalidad de una metodología de desarrollo es garantizar la eficacia y la eficiencia en el proceso de generación de software.”(Cuaresma, 2008)

Proceso Unificado Ágil (AUP)

El Proceso Unificado Ágil de Scott Ambler o *Agile Unified Process* (AUP) es una versión simplificada del Proceso Unificado Racional o *Rational Unified Process* (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP.

El AUP aplica técnicas ágiles incluyendo:

- Desarrollo dirigido por pruebas
- Modelado ágil
- Gestión de cambios ágil
- Refactorización de base de datos para mejorar la productividad

Al no existir una metodología de software universal, ya que toda metodología debe ser adaptada a las características de cada proyecto exigiéndose así que el proceso sea configurable, se decide hacer una variación de la metodología AUP, de forma tal que se adapte al ciclo de vida definido para la actividad productiva de la UCI.

De las 4 fases que propone AUP (Inicio, Elaboración, Construcción, Transición) se decide para el ciclo de vida de los proyectos de la UCI mantener la fase de Inicio, pero modificando el objetivo de la misma, se unifican las restantes 3 fases de AUP en una sola, llamada Ejecución y se agrega la fase de Cierre.

Capítulo 1: Fundamentación teórica

AUP propone 7 disciplinas (Modelo, Implementación, Prueba, Despliegue, Gestión de configuración, Gestión de proyecto y Entorno), se decide para el ciclo de vida de los proyectos de la UCI tener 8 disciplinas, pero a un nivel más atómico que el definido en AUP. Los flujos de trabajos: Modelado de negocio, Requisitos y Análisis y diseño en AUP están unidos en la disciplina Modelo, en la variación para la UCI se consideran a cada uno de ellos disciplinas. Se mantiene la disciplina Implementación, en el caso de Prueba se desagrega en 3 disciplinas: Pruebas Internas, de Liberación y Aceptación y la disciplina Despliegue se considera opcional. Las restantes 3 disciplinas de AUP asociadas a la parte de gestión para la variación UCI se cubren con las áreas de procesos que define CMMI-DEV v1.3 para el nivel 2, serían CM (Gestión de la configuración), PP (Planeación de proyecto) y PMC (Monitoreo y control de proyecto).

Se selecciona AUP, debido a la familiarización que se posee con los artefactos que son manejados en las diferentes fases de la misma. Entre las ventajas que brinda está en su apoyo a un proceso formalizado evitando las improvisaciones, tiene bien definido los roles y fases, es ágil estando basada en un proceso iterativo evolutivo, incrementa la productividad y facilita el trabajo en proyectos de pequeña envergadura.

1.6.2 Lenguajes de programación y etiquetado

PHP 5.4.12

PHP es un lenguaje de programación interpretado diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica. PHP es un acrónimo recursivo que significa PHP *Hypertext Pre-processor*.

Entre sus principales características destacan su potencia, alto rendimiento, facilidad de aprendizaje y escasez de consumo de recursos. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. Lo mejor de usar PHP es que es extremadamente simple para principiantes, pero a su vez, ofrece características avanzadas para los programadores profesionales.

JavaScript

Capítulo 1: Fundamentación teórica

JavaScript es un lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Se utiliza principalmente en su forma del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF y aplicaciones de escritorio es también significativo. Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web.

Maneja objetos dentro de la página web y sobre ese objeto es posible definir diferentes eventos. Dichos objetos facilitan la programación de páginas interactivas, a la vez que se evita la posibilidad de ejecutar comandos que puedan ser peligrosos para la máquina del usuario, tales como formateo de unidades y modificación de archivos. Es dinámico y responde a eventos en tiempo real; eventos como presionar un botón, pasar el puntero del mouse sobre un determinado texto o el simple hecho de cargar la página o caducar un tiempo. Es posible cambiar totalmente el aspecto de la página al gusto del usuario, evitando tener en el servidor una página para cada gusto o hacer cálculos en base a variables cuyo valor es determinado por el usuario.

HTML 5.0

Lenguaje de Marcado de Hipertexto (HTML, de sus siglas en *inglés* *HyperText Markup Language*), es el lenguaje básico para la elaboración de páginas web. Define comandos, marcas o etiquetas que permiten delimitar la estructura lógica de un documento web (Lenguajes de Programación, 2009).

Se utilizará el lenguaje HTML en su versión v5.0 ya que ofrece nuevas características con respecto a las versiones anteriores de HTML que proporcionan no solo una amplia compatibilidad de medios, sino también una compatibilidad mejorada para la creación de aplicaciones web que pueden interactuar de una manera más sencilla y efectiva con el usuario, sus datos locales y los servidores.

Hojas de Estilos en Cascada

Hojas de Estilos en Cascada (CSS, de sus siglas en *inglés* *Cascading Style Sheets*), es un lenguaje que se utiliza para definir el estilo de presentación que tendrá un documento HTML o XML y por extensión XHTML. Estas hojas de estilo se utilizan para separar la estructura del documento de la presentación del mismo (Mir, 2012).

Capítulo 1: Fundamentación teórica

Se utilizará CSS en su versión 3.0. ya que se obtiene un mayor control de la presentación del sitio al poder tener todo el código CSS reunido en uno, lo que facilita su modificación y se logra hacer mucho más legible el código HTML al tener el código CSS aparte.

XML

XML no es un lenguaje en particular sino que sirve de marco para definir lenguajes para diferentes necesidades. Algunos ejemplos son XHTML, XSLT, SOAP. Su diseño persigue enfatizar la simplicidad y usabilidad a través de Internet. Su formato es basado en caracteres de texto con soporte para todos los idiomas. Originalmente diseñado para afrontar el reto que representa la publicación de información electrónica a gran escala. XML ha jugado un gran papel en el intercambio de información en la web y otras esferas. Su objetivo es facilitar la representación, almacenamiento y transmisión de información por parte de aplicaciones informáticas, computadoras y medios de comunicación digital en general.

1.6.3 Herramienta CASE

La herramienta CASE seleccionada para el desarrollo de esta investigación es Visual Paradigm 8.0 (VP), para dar continuidad al trabajo realizado anteriormente además de tener en cuenta las características que presenta la herramienta y ambiente de trabajo. Visual Paradigm es una herramienta multiplataforma, permite esbozar todos los diagramas de clases necesarios, tiene integración con diferentes entornos de desarrollo, generación de código y documentación. VP posee un diseño de interfaz gráfica enfocado a facilitar la interacción y el trabajo del usuario. La herramienta también proporciona abundantes tutoriales, demostraciones interactivas y proyectos de UML. Permite invertir código fuente de programas, archivos ejecutables y binarios en modelos UML al instante, creando de manera simple toda la documentación.

Provee una estructura de proyecto jerarquizada donde se pueden modelar las diferentes perspectivas del sistema y sus artefactos en cada fase. Incluye herramientas de animación para simular el funcionamiento del modelado que se construye, permite por otra parte: el análisis de impacto, administración de tareas, diseño colaborativo y generación de código en múltiples lenguajes a partir de clases. (Visual Paradigm, 2015)

Se selecciona la herramienta case Visual Paradigm 8.0 ya que la misma soporta aplicaciones web, apoya todo lo básico en cuanto a artefactos generados en las etapas de definición de requerimientos y de especificación de componentes. Además permite la

Capítulo 1: Fundamentación teórica

generación de código e ingeniería inversa: brinda la posibilidad de generar código a partir de los diagramas, así como obtener los diagramas a partir del código.

1.6.4 Servidor Web

“Un servidor web puede ser tanto un ordenador de grandes dimensiones y capacidad como un programa informático que utiliza el protocolo de comunicaciones http para recibir las peticiones de información de un programa cliente (navegador) en el ordenador del usuario.”(Consumoteca, 2009)

Apache 2.4.7

Apache es un servidor *HyperText Transfer Protocol* (HTTP) de código abierto que por su facilidad de configuración, robustez y estabilidad se encuentra desplegado en millones de servidores. Es multiplataforma, haciéndolo prácticamente universal, tecnología gratuita de código abierto, patentado a través de la licencia Apache, además es extensible. (apache.org, 2013).

Constituye el servidor HTTP más popular y usado, lo que permite que sea fácil de adquirir y brinda una amplia fuente de ayuda. Muestra, entre otras características, mensajes de errores altamente configurables, posee bases de datos de autenticación y negociado de contenido, facilita la utilización de Perl, PHP, Java y otros lenguajes *script*. (apache.org, 2013).

1.6.5 Sistema gestor de bases de datos

Los sistemas gestores de bases de datos se pueden definir según (Navathe, 1997) como “un sistema de software de propósito general que facilita el proceso de definir, construir y manipular bases de datos para diversas aplicaciones”.

- Definir: especificar los tipos de datos, las estructuras y las restricciones de los datos.
- Construir: guardar los datos mismos en algún medio de almacenamiento controlado por el SGBD.
- Manipular: consultar y actualizar la información almacenada.

PostgreSQL 9.2

PostgreSQL es un sistema gestor de bases de datos relacional orientado a objetos. Es de código abierto, brinda un control de concurrencia multiversión (MVCC por sus siglas en

Capítulo 1: Fundamentación teórica

inglés) que permite trabajar con grandes volúmenes de datos; soporta gran parte de la sintaxis SQL y cuenta con un extenso grupo de enlaces con lenguajes de programación.

Como características de este gestor se encuentran (Quiñones, 2008):

- La atomicidad, esta propiedad asegura que la operación se haya realizado o no, así permite que ante un fallo del sistema la acción que se ejecutaba no queda a medias.
- Consistencia en la propiedad, asegura que las tareas una vez empezadas se finalicen correctamente y completas.
- Aislamiento en la propiedad, asegura que una operación no pueda afectar a las otras.
- Durabilidad, asegura que una vez realizadas las operaciones, aunque existan fallas en el sistema estas se mantienen persistentes.
- Se ejecuta en casi todos los sistemas operativos, tales como Linux y Windows.
- Diversidad de documentación, amplia, organizada, libre y pública.

Se decide utilizar PostgreSQL, debido a que es un software con licencia libre, se dispone de una amplia documentación en línea y en la UCI y por las propiedades anteriormente mencionadas.

1.6.6 Entorno de desarrollo integrado (IDE)

Un entorno de desarrollo integrado, llamado también IDE (por sus siglas en inglés *Integrated Development Environment*), es un programa informático compuesto por un conjunto de herramientas de programación. Es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI).

NetBeans 8.0

El proyecto NetBeans consiste en un IDE de código abierto y una plataforma de aplicaciones que permiten a los desarrolladores crear aplicaciones web, empresariales, de escritorio y móviles, utilizando la lenguajes como Java, PHP, JavaScript y Ajax y C/C++.

El IDE NetBeans es un entorno de desarrollo, está escrito en Java pero puede servir para otros lenguajes de programación. Es un producto libre y gratuito sin restricciones de uso, disponible para Windows, Mac, Linux y Solaris. (Domínguez-Dorado, 2005)

Capítulo 1: Fundamentación teórica

Se selecciona NetBeans debido a que es un IDE, robusto y que brinda facilidades para el desarrollo de aplicaciones web utilizando los lenguajes requeridos para el desarrollo de la aplicación, presenta abundante documentación y el equipo de desarrollo tiene experiencia trabajando con el mismo.

1.6.7 Framework de desarrollo Symfony 2

Symfony es una biblioteca de clases de cohesión escrito en PHP. Proporciona una arquitectura, componentes y herramientas a los desarrolladores para construir aplicaciones web complejas más rápido. Symfony es un *framework* diseñado para optimizar el desarrollo de aplicaciones web por medio de varias características claves. Separa las reglas de una aplicación web de negocios, lógica del servidor, y las vistas de presentación. Contiene numerosas herramientas y clases dirigidas a acortar el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes para que el desarrollador pueda concentrarse por completo en los detalles de una aplicación. Es compatible con la mayoría de los motores de bases de datos disponibles, como MySQL, PostgreSQL, Oracle, y Microsoft SQL Server. (Zaninotto, 2007)

Se seleccionó como *framework* de desarrollo Symfony 2 ya que proporciona soluciones prefabricadas para los problemas más comunes, como son, la atención de peticiones, formularios e interacción con base de datos. También ofrece una estructura clara y organizada a varios niveles, como la estructura de directorios y la separación por capas, proporciona una buena seguridad para las aplicaciones y cuenta con una vasta documentación tanto en internet como en la UCI.

1.6.8 Framework JavaScript JQuery 1.9.2

Jquery consiste en un único fichero JavaScript que contiene las funcionalidades comunes de *Document Object Model* (DOM) o Modelo de Objetos del Documento, eventos, efectos, *Asynchronous JavaScript And XML* (AJAX) o Java asincrónico y XML, así como *JavaScript Object Notation* (JSON) o Notación de objetos JavaScript. Es una biblioteca o *framework* de JavaScript. Permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la tecnología AJAX a páginas Web.

Jquery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las

Capítulo 1: Fundamentación teórica

funciones propias de esta biblioteca se logran buenos resultados en poco tiempo y espacio.

1.6.9 Framework CSS Bootstrap 2.3

Es el *framework* para CSS más popular que existe para el desarrollo web, posee una amplia documentación, haciéndose extensible su uso. Provee de forma fácil las mejores prácticas en el diseño de plantillas, donde con este se generan vistas muy agradables y ligeras para el cliente. Es de código abierto, con una comunidad que lo mantiene a través de GitHub. (Bootstrap, 2015)

1.7 Conclusiones parciales

Luego de haber realizado el estudio para dar solución al problema planteado en la presente investigación se obtuvo como resultado la necesidad de elaboración de un módulo para la creación de diseños de aprendizaje basándose en la especificación IMS-LD para la herramienta de autor web CRODA 3.0.

Las investigaciones realizadas sobre el estado actual de las metodologías de desarrollo de software han permitido adoptar como se expuso anteriormente los artefactos propuestos por la metodología AUP en su variación para la UCI. Para el proceso de análisis se hará uso de la herramienta CASE Visual Paradigm 8.0.

Los lenguajes seleccionados para la implementación, teniendo en cuenta sus ventajas, características y correspondencia con el entorno de desarrollo, fueron: HTML 5, CSS 3, JavaScript, XML y PHP 5.4.12. Una vez analizadas las herramientas, tecnologías y marcos de trabajo más usados en el desarrollo de aplicaciones web se determinó el uso de los que se enuncian a continuación: NetBeans 8.0, PostgreSQL, Apache, jQuery, Bootstrap 2.3, Symfony 2.3.

Capítulo 2: Análisis y diseño

Capítulo 2: Análisis y diseño

Introducción

La metodología AUP-UCI define en cada una de sus fases un grupo de artefactos que le brindan al equipo de desarrollo una comprensión clara del sistema a desarrollar, al tener un enfoque ágil permite que el equipo elabore la documentación necesaria que facilite la comunicación entre los desarrolladores y las partes interesadas. En este capítulo se presentarán los siguientes artefactos: la especificación de requisitos funcionales y no funcionales que tendrá la propuesta de solución y el Modelado de Diseño. También se describen los patrones arquitectónicos a utilizar en la propuesta solución así como los patrones de diseño seleccionados.

2.1 Descripción de la propuesta solución

El objetivo del módulo a desarrollar como propuesta solución es permitir la creación de diseños de aprendizaje basándose en la especificación IMS-LD hasta el nivel A, el mismo permitirá la gestión de los diseños de aprendizaje, crearlos, listarlos ,editarlos y eliminarlos. También hará posible la gestión de los elementos contenidos en el nivel A de la especificación.

La principal diferencia de la propuesta solución con la versión anterior del módulo, presente en CRODA 2.0, es que los elemento del nivel A de la especificación serán tratados como objetos, con propiedades y funciones que se pueden realizar sobre ellos, como son la inserción de elementos que van contenidos dentro de cada uno, modificar sus datos y eliminarlos, así como moverlos por el ambiente de trabajo y cambiar su tamaño a consideración del usuario con el objetivo de hacer más fácil la creación del diseño de aprendizaje.

2.2 Requisitos del sistema

La definición de requisitos del sistema permite establecer las capacidades del mismo y sobre qué restricciones o condiciones debe este trabajar. Los mismos son planteados con el fin de lograr un entendimiento claro entre el cliente y el equipo de desarrollo, en aras de lograr satisfacer sus verdaderas necesidades.

2.2.1 Requisitos funcionales

- **RF 1** Crear diseño de aprendizaje

Capítulo 2: Análisis y diseño

- **RF 2** Listar diseños de aprendizaje en edición
- **RF 3** Editar diseño de aprendizaje
- **RF 4** Eliminar diseño de aprendizaje
- **RF 5** Crear objetivos generales
- **RF 6** Editar objetivos generales
- **RF 7** Eliminar objetivos generales
- **RF 8** Crear prerrequisitos
- **RF 9** Editar prerrequisitos
- **RF 10** Eliminar prerrequisitos
- **RF 11** Crear roles
- **RF 12** Editar roles
- **RF 13** Eliminar roles
- **RF 14** Crear ambientes
- **RF 15** Editar ambientes
- **RF 16** Eliminar ambientes
- **RF 17** Crear actividades de aprendizaje
- **RF 18** Editar actividades de aprendizaje
- **RF 19** Eliminar actividades de aprendizaje
- **RF 20** Crear actividades de apoyo
- **RF 21** Editar actividades de apoyo
- **RF 22** Eliminar actividades de apoyo
- **RF 23** Adicionar parte del rol
- **RF 24** Editar parte del rol

Capítulo 2: Análisis y diseño

- **RF 25** Eliminar parte del rol
- **RF 26** Adicionar acto
- **RF 27** Editar acto
- **RF 28** Eliminar acto
- **RF 29** Adicionar obra
- **RF 30** Editar obra
- **RF 31** Eliminar obra
- **RF 32** Completar método
- **RF 33** Adicionar recurso
- **RF 34** Eliminar recurso
- **RF 35** Mostrar vista estructural

2.2.2 Requisitos no funcionales

Los requisitos no funcionales son propiedades o cualidades que el producto debe tener. Es necesario pensar en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Estos requisitos pueden marcar la diferencia entre un producto bien aceptado y otro con poca aceptación. (Cruz Castro, 2011)

Para especificar los requerimientos del software se definieron una serie de categorías que engloban las cualidades que debe mantener un sistema para su correcto funcionamiento y una prestación adecuada de sus servicios.

Usabilidad

- El sistema debe poseer una interfaz fácil de utilizar para cualquier tipo de usuarios con conocimientos básicos de informática y del manejo de ordenadores.

Accesibilidad

- El sistema debe estar disponible desde cualquier estación de trabajo conectada a la red y permitir su administración de forma remota.

Capítulo 2: Análisis y diseño

Apariencia

- El sistema debe tener una interfaz con colores adecuados a la línea de proyectos XAUCE, establecida por la estrategia marcaria de los productos UCI.
- La interfaz debe ser de fácil comprensión en su funcionamiento permitiendo la utilización del sistema sin mucho entrenamiento.
- La interfaz contará con menús que faciliten y aceleren su utilización.
- Interfaces uniformes y con los mismos colores y diseños.

Disponibilidad.

- La aplicación debe estar disponible en todo momento. Se les garantizará a los usuarios el acceso a la información.

Soporte

- La aplicación debe ser implementada bajo tecnología web, puesto que la misma será accedida a través de Internet.
- Ejecutarse sobre cualquier navegador, siendo como mínimo compatible con:
 - Mozilla Firefox 7.0 y superior.
 - Opera 10.0.0 y superior.
 - Chrome 7.0 y superior.

Restricciones de diseño e implementación

- Se aplicará la programación orientada a objetos.
- El *framework* de desarrollo que se utilizará es: Symfony v2.3.7
- Como IDE se empleará NetBeans v8.0.
- Se empleará como sistema gestor de bases de datos PostgreSQL v9.2
- Como servidor web se empleará Apache v2.4.

2.3 Descripción de requisitos

Capítulo 2: Análisis y diseño

La descripción de requisitos refleja la respuesta que cabe esperar del sistema, la prioridad y la complejidad las cuales son clasificadas como Baja, Media y Alta dependiendo del orden de prioridad y complejidad respectivamente.

En la siguiente tabla se muestra la descripción de los requisitos para el módulo Diseño de aprendizaje:

Tabla 1: Descripción de los requisitos

N.	Nombre	Descripción	Prioridad para el cliente	Complejidad	Referencias cruzadas
RF 1	Crear diseño de aprendizaje	El sistema debe permitir crear un diseño de aprendizaje a partir de los siguientes datos: <ul style="list-style-type: none">• Nombre• Descripción• Palabras• Idioma• Método• Usuario	Alta	Media	N/A
RF 2	Listar Diseños de aprendizaje en edición	El sistema debe permitir listar los diseños de aprendizaje en edición.	Alta	Media	N/A
RF 3	Editar Diseño de aprendizaje	El sistema debe permitir editar los diseños de aprendizaje en edición.	Alta	Alta	N/A
RF 4	Eliminar	El sistema debe permitir eliminar los	Alta	Alta	N/A

Capítulo 2: Análisis y diseño

	Diseño de aprendizaje	diseños de aprendizaje en edición.			
RF 5	Crear objetivos generales	El sistema debe permitir añadir los objetivos generales del diseño de aprendizaje los cuales contendrán: <ul style="list-style-type: none"> • Nombre • Recurso 	Alta	Media	N/A
RF 6	Editar objetivos generales	El sistema debe permitir editar los objetivos generales de un diseño de aprendizaje.	Alta	Alta	N/A
RF 7	Eliminar objetivos generales	El sistema debe permitir eliminar los objetivos generales de un diseño de aprendizaje.	Alta	Alta	N/A
RF 8	Crear prerrequisitos	El sistema debe permitir añadir los prerrequisitos del diseño de aprendizaje los cuales contendrán: <ul style="list-style-type: none"> • Nombre • Recurso 	Alta	Media	N/A
RF 9	Editar prerrequisitos	El sistema debe permitir editar los prerrequisitos de un diseño de aprendizaje.	Alta	Alta	N/A
RF 10	Eliminar prerrequisitos	El sistema debe permitir eliminar los prerrequisitos de un diseño de aprendizaje.	Alta	Alta	N/A
RF 11	Crear roles	El sistema debe permitir crear roles con los siguientes datos:	Alta	Alta	N/A

Capítulo 2: Análisis y diseño

		<ul style="list-style-type: none"> • Nombre • Tipo de rol, que puede ser profesor o aprendiz • Responsabilidad que puede ser exclusivamente en funciones o no exclusivamente • Creador que puede ser permitido o no permitido • Máximo del rol • Mínimo del rol • Lista de informaciones del rol <p>El sistema debe permitir listar, editar y eliminar los roles creados.</p>			
RF 12	Editar roles	El sistema debe permitir editar los roles del diseño de aprendizaje.	Alta	Alta	N/A
RF 13	Eliminar roles	El sistema debe permitir eliminar los roles del diseño de aprendizaje.	Alta	Alta	N/A
RF 14	Crear ambientes.	El sistema debe permitir crear un ambiente en el diseño de aprendizaje.	Alta	Alta	N/A
RF 15	Editar ambientes	<p>El sistema debe permitir modificar el nombre de un ambiente.</p> <p>El sistema debe permitir añadir y modificar un objeto de aprendizaje a un ambiente que tendrá como datos:</p>	Alta	Alta	N/A

Capítulo 2: Análisis y diseño

	<ul style="list-style-type: none">• Nombre• Tipo, que puede ser: objeto de conocimiento, objeto de herramienta, objeto test y otros• Una lista con los recursos asociados al objeto de aprendizaje <p>El sistema debe permitir añadir y modificar una conferencia a un ambiente con los siguientes datos:</p> <ul style="list-style-type: none">• Nombre• Administrador, que debe ser un rol creado previamente• Moderador que debe ser un rol creado previamente.• Tipo que puede ser asíncrono, síncrono o anuncio• Una lista con los roles participantes• Una lista con los roles observadores• Una lista con los recursos asociados a la conferencia <p>El sistema debe permitir añadir y modificar un correo a un ambiente con los siguientes datos:</p>			
--	---	--	--	--

Capítulo 2: Análisis y diseño

	<ul style="list-style-type: none">• Nombre• Tipo que puede ser personal en el rol o todos en el rol• Rol, que debe ser un rol creado previamente <p>El sistema debe permitir añadir y modificar un índice y búsqueda a un ambiente con los siguientes datos:</p> <ul style="list-style-type: none">• Nombre• Tipo de elemento, que puede ser objeto de aprendizaje, conferencia, correo electrónico, actividad o estructura de actividades• Elemento, que debe contener los elementos que cumplan con el tipo de elementos seleccionado anteriormente.• Clase del índice.• Tipo de búsqueda, que puede ser Texto libre de búsqueda, índice con referencia e índice sin referencia. <p>El sistema debe permitir añadir y modificar una referencia a un ambiente, la cual referenciará un ambiente creado anteriormente.</p>			
--	--	--	--	--

Capítulo 2: Análisis y diseño

RF 16	Eliminar ambientes	El sistema debe permitir eliminar un ambiente creado previamente.	Alta	Alta	N/A
RF 17	Crear actividades de aprendizaje	El sistema debe permitir crear una actividad de aprendizaje del diseño de aprendizaje.	Alta	Alta	N/A
RF 18	Editar actividades de aprendizaje	<p>El sistema debe permitir añadir y modificar una descripción a una actividad de aprendizaje con los datos:</p> <ul style="list-style-type: none"> • Nombre • Recurso <p>El sistema debe permitir añadir y modificar objetivos a una actividad de aprendizaje con los datos:</p> <ul style="list-style-type: none"> • Nombre • Recurso <p>El sistema debe permitir añadir y modificar prerrequisitos a una actividad de aprendizaje con los datos:</p> <ul style="list-style-type: none"> • Nombre • Recurso <p>El sistema debe permitir añadir y modificar condiciones de completamiento a la actividad de</p>	Alta	Alta	N/A

Capítulo 2: Análisis y diseño

		<p>aprendizaje que puede ser:</p> <ul style="list-style-type: none"> • A elección del usuario • Por tiempo <p>El sistema debe permitir agregar y modificar una retroalimentación a la actividad de aprendizaje.</p>			
RF 19	Eliminar actividades de aprendizaje	El sistema debe permitir eliminar una actividad de aprendizaje creada anteriormente.	Alta	Alta	N/A
RF 20	Crear actividades de apoyo	El sistema debe permitir crear una actividad de apoyo del diseño de aprendizaje.	Alta	Alta	N/A
RF 21	Editar actividades de apoyo	<p>El sistema debe permitir añadir y modificar una descripción a una actividad de apoyo con los datos:</p> <ul style="list-style-type: none"> • Nombre • Recurso <p>El sistema debe permitir añadir y modificar referencia de roles a una actividad de apoyo con los datos:</p> <ul style="list-style-type: none"> • Identificador del rol • El sistema debe permitir añadir y modificar condiciones de completamiento a la actividad de apoyo que puede ser: 	Alta	Alta	N/A

Capítulo 2: Análisis y diseño

		<ul style="list-style-type: none"> • A elección del usuario • Por tiempo <p>El sistema debe permitir agregar y modificar una retroalimentación a la actividad de apoyo.</p>			
RF 22	Eliminar actividades de apoyo	El sistema debe permitir eliminar una actividad de apoyo creada anteriormente.	Alta	Alta	N/A
RF 23	Crear parte del rol	El sistema debe permitir crear una parte del rol.	Alta	Alta	N/A
RF 24	Editar parte del rol	El sistema debe permitir editar una parte del rol con los siguientes elementos: <ul style="list-style-type: none"> • Nombre 	Alta	Alta	N/A
RF 25	Eliminar parte del rol	El sistema debe permitir eliminar una parte del rol del diseño de aprendizaje.	Alta	Alta	N/A
RF 26	Crear acto	El sistema debe permitir crear un acto del diseño de aprendizaje.	Alta	Alta	N/A
RF 27	Editar acto	El sistema debe permitir añadir y modificar una descripción a un acto con los datos:	Alta	Alta	N/A

Capítulo 2: Análisis y diseño

		<ul style="list-style-type: none"> • Nombre • Recurso <p>El sistema debe permitir añadir y modificar condiciones de completamiento a la actividad de apoyo que puede ser:</p> <ul style="list-style-type: none"> • Al terminar las partes del rol seleccionadas • Por tiempo <p>El sistema debe permitir agregar y modificar una retroalimentación al acto</p>			
RF 28	Eliminar acto	El sistema debe permitir eliminar un acto creado anteriormente.	Alta	Alta	N/A
RF 29	Crear obra	El sistema debe permitir crear una obra en el diseño de aprendizaje.	Alta	Alta	N/A
RF 30	Editar obra	<p>El sistema debe permitir añadir y modificar una descripción a una obra con los datos:</p> <ul style="list-style-type: none"> • Nombre • Recurso <p>El sistema debe permitir añadir y modificar condiciones de completamiento a la obra que puede ser:</p>	Alta	Alta	N/A

Capítulo 2: Análisis y diseño

		<ul style="list-style-type: none"> • Al terminar el último acto. • Por tiempo <p>El sistema debe permitir agregar y modificar una retroalimentación a la actividad de apoyo.</p>			
RF 31	Eliminar obra	El sistema debe permitir eliminar una obra creada anteriormente.	Alta	Alta	N/A
RF 32	Completar método.	<p>El sistema debe permitir modificar las condiciones de completamiento al método que puede ser:</p> <ul style="list-style-type: none"> • Al terminar obras seleccionadas • Por tiempo <p>El sistema debe permitir modificar una retroalimentación al método.</p>	Alta	Alta	N/A
RF 33	Adicionar recurso.	El sistema debe permitir añadir un recurso al diseño de aprendizaje, el recurso puede ser un archivo cualquiera.	Alta	Alta	N/A
RF 34	Eliminar recurso.	El sistema debe permitir eliminar un recurso del diseño de aprendizaje.	Alta	Alta	N/A
RF 35	Mostrar vista estructural	El sistema debe mostrar la estructura del método en vista estructural con forma de árbol con la jerarquía perteneciente a los elementos del nivel A de la especificación IMS-LD.	Alta	Alta	N/A

Capítulo 2: Análisis y diseño

		La vista estructural debe actualizarse cada vez que se modifique algún elemento del método, al agregar un elemento, renombrarlo o eliminarlo.			
--	--	---	--	--	--

2.4 Especificación de requisitos

A continuación se muestra la especificación de requisitos correspondientes a diseño de aprendizaje y acto. Para tener información del resto de las especificaciones consultar los anexos.

Tabla 2: Especificación de requisito Crear diseño de aprendizaje

Precondiciones	El usuario debe estar autenticado.
Flujo de eventos	
Flujo básico: Crear diseño de aprendizaje.	
1	El sistema muestra la lista de los diseños de aprendizaje existentes.
2	Se selecciona el botón Crear nuevo diseño de aprendizaje.
3	Se muestra un formulario con los siguientes campos a llenar: <ul style="list-style-type: none"> • Nombre • Descripción • Palabras Claves • Idioma
4	Se selecciona la opción Crear nuevo diseño de aprendizaje.
5	El sistema valida los campos insertados
6	Si los datos son correctos el sistema crea el diseño de aprendizaje.
7	Se muestra la vista de creación del contenido del diseño de aprendizaje.
Post-condiciones	
1	Se crea un diseño de aprendizaje.
Flujos alternativos	
Flujo alternativo 5.a Datos incorrectos	
1	El sistema señala los datos incorrectos y permite corregirlos.
2	Se corrigen los datos.
3	Volver al paso 4 del flujo básico.
Post-condiciones	

Capítulo 2: Análisis y diseño

1	N/A	
Flujo alternativo 4.a El usuario selecciona la opción Regresar a la lista		
1	El sistema muestra la lista de diseños de aprendizaje	
2	Concluye el requisito.	
Post-condiciones		
1	No se crea el diseño de aprendizaje.	
2	Se muestra la lista de diseños de aprendizaje disponibles.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: Visibles en la interfaz: <ul style="list-style-type: none"> • Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Capítulo 2: Análisis y diseño

Crear diseño de aprendizaje

Nombre

Descripción

Palabras

Idioma

Figura 4: Prototipo interfaz gráfica de usuario

Formatos de entrada/salida

N/A

Entradas

N/A

Salidas

N/A

Tabla 3: Especificación de requisito Crear acto

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista
-----------------------	--

Capítulo 2: Análisis y diseño

	de creación del contenido del diseño de aprendizaje y se ha creado una obra.	
Flujo de eventos		
Flujo básico: Crear acto.		
1	Se selecciona la opción método.	
2	Se selecciona el elemento acto	
3	Se da un clic en una obra creada.	
4	El sistema agrega el acto a la obra cliqueada.	
Post-condiciones		
1	Se Agrega el acto a la obra.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

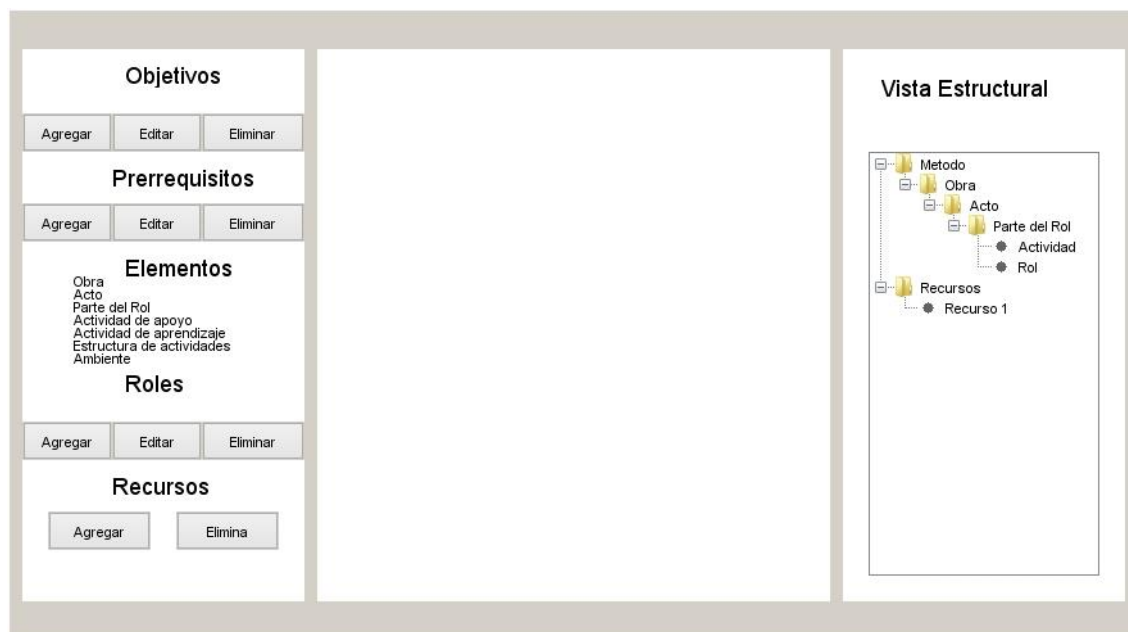


Figura 5: Prototipo interfaz gráfica de usuario

Capítulo 2: Análisis y diseño

Formatos de entrada/salida

N/A

Entradas

N/A

Salidas

N/A

Tabla 4: Especificación de requisito Editar acto

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el acto ha sido creado.
Flujo de eventos	
Flujo básico: Editar acto.	
1	Se da clic derecho sobre el acto a editar.
2	Se selecciona la opción editar información.
3	Se muestra un formulario con los siguientes campos con los datos del acto: <ul style="list-style-type: none">• Descripción del acto.• Completamiento del acto.• Retroalimentación del acto.
4	Se editan los datos.
5	Se selecciona la opción Aceptar.
6	El sistema valida los campos insertados.
7	Si los datos son correctos el sistema edita el acto.
8	Concluye el requisito.
Post-condiciones	
1	Se edita el acto.
Flujos alternativos	
Flujo alternativo 6.a Datos incorrectos.	
1	El sistema señala los datos incorrectos y permite corregirlos.

Capítulo 2: Análisis y diseño

2	Se corrigen los datos.	
3	Volver al paso 5 del flujo básico.	
Post-condiciones		
1	N/A	
Flujo alternativo 5.a Se selecciona el botón cancelar.		
1	Concluye el requisito.	
Post-condiciones		
1	No se edita el acto.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Capítulo 2: Análisis y diseño

Editar Información del acto

Descripción

Agregar Eliminar

Completar Acto

Al terminar los roles Por tiempo

[Dropdown menu]

Eliminar

Años [input] Días [input] Minutos [input]
Meses [input] Horas [input] Segundos [input]

Retroalimentación

[Text input]

Aceptar Cancelar

Figura 6: Prototipo interfaz gráfica de usuario

Formatos de entrada/salida

N/A

Entradas

N/A

Salidas

N/A

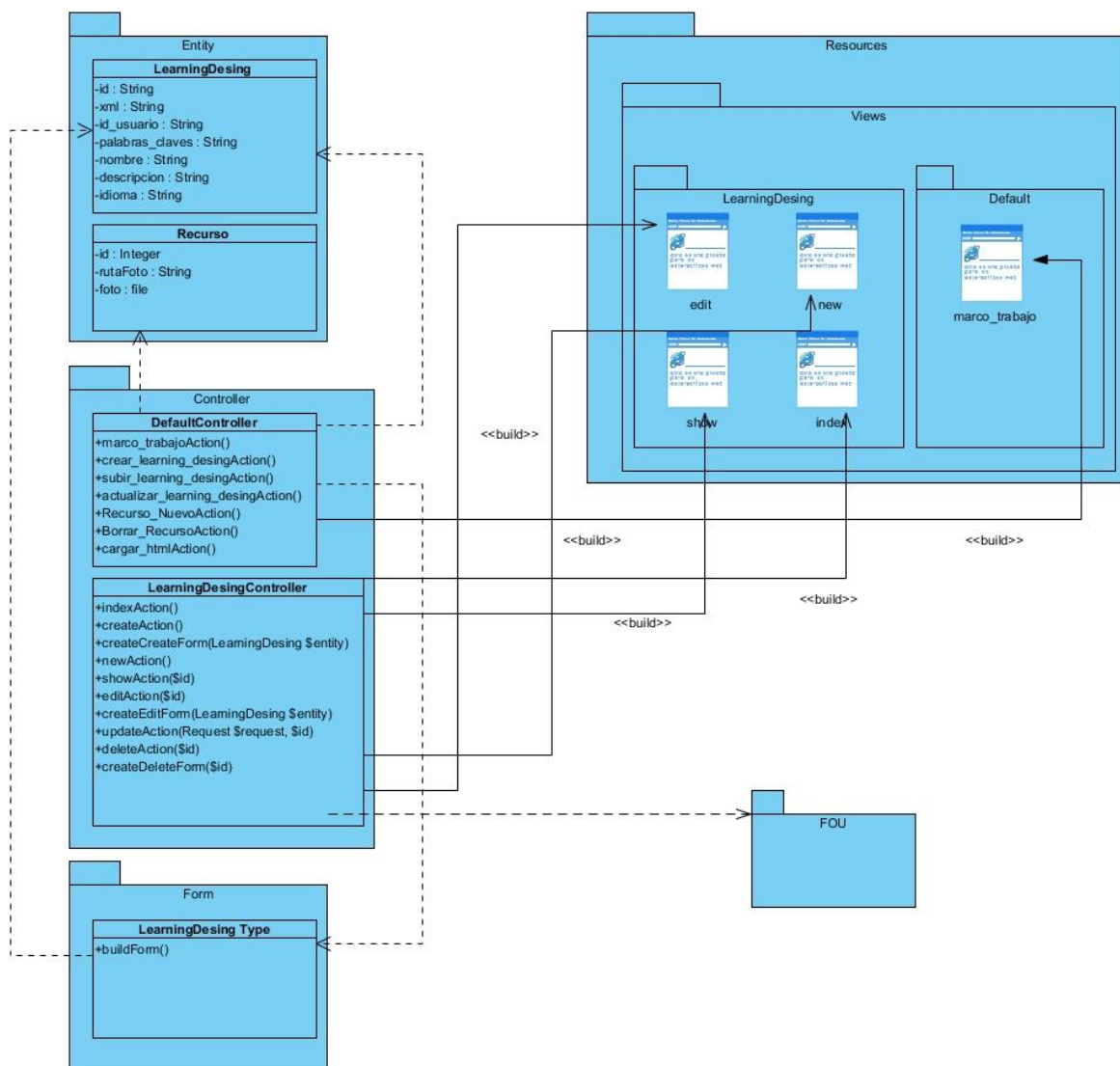
2.5 Modelo de diseño

Capítulo 2: Análisis y diseño

El modelo de diseño crea una representación o modelo de software enfocado en la representación de los datos, las funciones y el comportamiento requerido, permite al ingeniero de *software* modelar el sistema o producto que se va a construir pudiéndose evaluar su calidad y mejorarlo antes de generar el código, dándole alta calidad al *software*. Con este se obtiene una representación arquitectónica de interfaz y despliegue del sistema. (Pressman, 2002)

2.5.1 Diagramas de clases del diseño

Se presenta a continuación el diagrama de clases de diseño del módulo Diseño de aprendizaje, donde se observan los paquetes que agrupan las clases participantes y sus respectivas relaciones.



Capítulo 2: Análisis y diseño

Figura 7: Diagrama de Clase de Diseño de aprendizaje

A continuación se muestran las descripciones de las clases que componen el diagrama de clases de diseño mostrado:

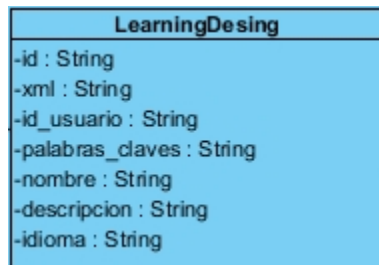


Figura 8: Clase entidad LearningDesign

LearningDesign: es la clase encargada de modelar la entidad LearningDesign en la base de datos.

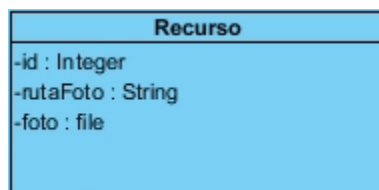


Figura 9: Clase entidad Recurso

Recurso: es la clase encargada de modelar la entidad Recurso en la base de datos.

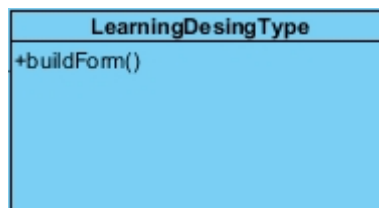


Figura 10: Clase LearningDesignType

LearningDesignType: es la clase encargada de construir el formulario para insertar los datos a una entidad LearningDesign.

Capítulo 2: Análisis y diseño

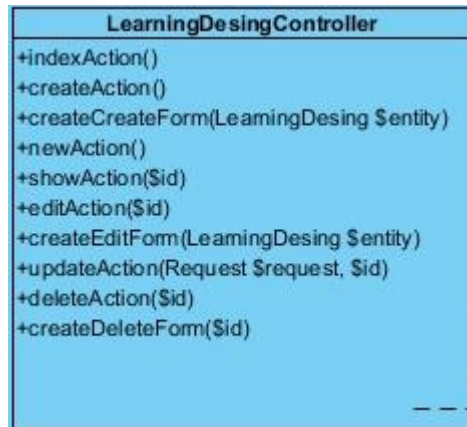


Figura 11: Clase Controladora LearningDesignController

LearningDesignController: es la clase encargada de realizar las funciones predeterminadas sobre una entidad LearningDesign como son la creación, edición.

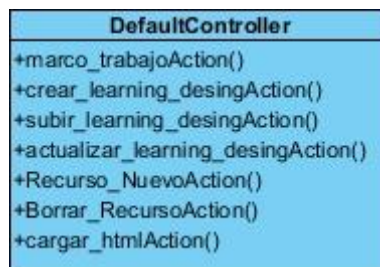


Figura 12: Clase controladora DefaultController

DefaultController: es la clase encargada de realizar las funciones necesarias sobre la entidad LearningDesign y Recurso para lograr el funcionamiento del módulo Diseño de aprendizaje.

2.6 Patrones de diseño

En la actualidad no se concibe la construcción de un sistema robusto, sin que exista en su diseño el uso de patrones, fundamentalmente en el diseño orientado a objetos donde existe un amplio repertorio y expresiones que guían el proceso de construcción de los sistemas, estos constituyen la solución de un problema determinado y se pueden aplicar en diferentes contextos, su finalidad no es expresar nuevas ideas en el diseño sino resolver los problemas mediante una solución ya probada y fiable. (Olivares, 2007)

Se decide emplear en el diseño del sistema patrones GRASP (Patrones Generales de Software para Asignar Responsabilidades, siglas en ingles) y GOF (Grupo de Cuatro,

Capítulo 2: Análisis y diseño

siglas en ingles), los cuales serán útiles para facilitar la implementación del sistema, posibilitar la reutilización de código y lograr un software con alto grado de mantenibilidad.

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones, define un grupo de principios fundamentales para diseñar eficazmente un software. (Craig, 1999) Para la propuesta de solución se utilizarán los siguientes.

Creador: se emplea cuando se le aplica la responsabilidad a una clase determinada de crear una o más instancias de otra, esto sucede en caso de que la clase creadora contenga, agregue, registre, utilice o posea datos de inicialización de objetos de alguna clase determinada. (Craig, 1999) En Symfony2 en la clase Controller se definen y ejecutan todas las acciones. En las acciones se crean los objetos de las clases que representan las entidades, evidenciando de este modo que la clase Controller es "creador" de dichas entidades.

Experto: se basa en asignar la responsabilidad a una clase que se encargará de ser experto de la información, la misma cuenta con la información necesaria para cumplir con la responsabilidad. (Craig, 1999) Al aplicarlo al sistema, este patrón permitirá identificar las clases que poseen la información requerida, para que luego estas puedan ser manipuladas por otras clases, esto se evidencia en las clases entidades, las cuales son expertas en la información de un concepto determinado.

Alta cohesión: consiste en asignar una responsabilidad de modo que la cohesión siga siendo alta, resolviendo el problema de mantener la complejidad en las clases dentro de los límites manejables, permite definir en las clases las funcionalidades relacionadas para que estas no realicen un trabajo enorme. (Craig, 1999) Este patrón permitirá que la implementación de las clases encargadas de ofrecer los servicios del sistema sean fáciles de comprender, reutilizar, mantener y poco susceptibles a cambios. En el sistema, la clase LearningDesignController solo manejará las funcionalidades realizadas sobre la entidad LearningDesign.

Controlador: asigna la responsabilidad del manejo de mensajes de los eventos de un sistema a una clase que represente el sistema global, organización, elemento activo del mundo real o manejador artificial de los eventos del sistema. (Craig, 1999) Dicho patrón será aplicado en el sistema mediante el uso de clases que se encargarán del manejo de

Capítulo 2: Análisis y diseño

las peticiones del usuario y para realizar operaciones con las clases entidades a fin de dar respuestas a los requisitos.

Bajo acoplamiento: el bajo acoplamiento es un principio que se debe recordar durante las decisiones de diseño: es la meta principal que es preciso tener siempre. Es un patrón evaluativo que el diseñador aplica al juzgar sus decisiones de diseño. Asigna una responsabilidad a una clase para mantener bajo acoplamiento, o sea, disminuir la dependencia entre clases, evitando que una modificación en alguna de ellas repercuta en gran medida en el resto, posibilitando además una mayor reutilización.

En el sistema la clase LearningDesignController hereda solamente de Controller, por lo que se garantiza que exista un grado moderado de acoplamiento entre las clases. Además, al no asociar las clases del modelo con las de la vista o el controlador, la dependencia entre las clases, en este caso, se mantiene baja.

Los patrones GOF permiten resolver problemas de diseño relacionados con la creación de instancias, permitiendo encapsular y abstraer las mismas (Gamma, 1995).

Singleton: garantiza que una clase tenga una única instancia y proporciona un punto de acceso global a esta, la propia clase es responsable de crear la única instancia y permite el acceso a través de alguno de sus métodos. Se implementa creando en la clase un método que crea una instancia del objeto sólo si todavía no existe alguna (Mühlrad, 2008).

2.7 Patrones Arquitectónicos

El patrón Modelo-Vista-Controlador (MVC) surge con el objetivo de reducir el esfuerzo de programación, necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos, a partir de estandarizar el diseño de las aplicaciones. El patrón MVC es un paradigma que divide las partes que conforman una aplicación en el modelo, las vistas y los controladores, permitiendo la implementación por separado de cada elemento, garantizando así la actualización y mantenimiento del software de forma sencilla y en un reducido espacio de tiempo. A partir del uso de frameworks basados en el patrón MVC se puede lograr una mejor organización del trabajo y mayor especialización de los desarrolladores y diseñadores. (González, 2012)

Este modelo de arquitectura presenta varias ventajas: (Catalani, 2007)

Capítulo 2: Análisis y diseño

- Separación clara entre los componentes de un programa; lo cual permite su implementación por separado.
- Conexión dinámica entre el modelo y sus vistas; se produce en tiempo de ejecución, no en tiempo de compilación.

El controlador es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el modelo, centra toda la interacción entre la vista y el modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del modelo o por alteraciones de la Vista. Interactúa con el modelo a través de una referencia al propio modelo. (González, 2012)

Un modelo puede tener diversas vistas, cada una con su correspondiente controlador, aunque es posible definir múltiples vistas para un único controlador. A continuación se detalla cada elemento. (Catalani, 2007)

El modelo es el responsable de:

- Acceder a la capa de almacenamiento de datos. Lo ideal es que el modelo sea independiente del sistema de almacenamiento.
- Define reglas de negocio (la funcionalidad del sistema). Es opcional, pues las reglas de negocio, pueden estar también en los controladores, directamente en las acciones.
- Notificará a las vistas los cambios que en los datos pueda producir un agente externo si se está ante un modelo.

El controlador es responsable de:

- Recibir los eventos de entrada.
- Contiene reglas de gestión de eventos, del tipo "Si Evento Z, entonces Acción W". Estas acciones pueden suponer peticiones al modelo o a las vistas.

Las vistas son responsables de:

- Recibir datos procesados por el controlador o del modelo y mostrarlos al usuario.
- Tienen un registro de su controlador asociado.

Capítulo 2: Análisis y diseño

- Pueden dar el servicio de "Actualización ()", para que sea invocado por el controlador o por el modelo cuando es un modelo activo que informa de los cambios en los datos producidos por otros agentes.

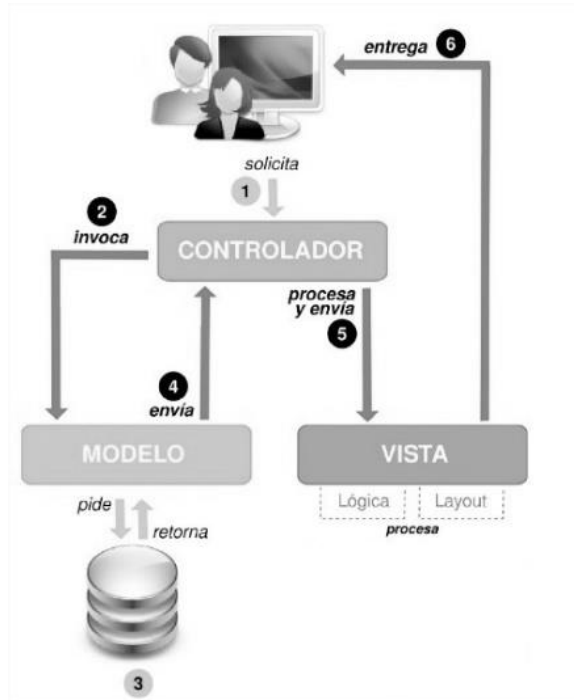


Figura 13: Funcionamiento del patrón Modelo Vista Controlador

Symfony 2 es un *framework* de PHP basado en la arquitectura Modelo-Vista-Controlador, por tanto, además de ser un patrón arquitectónico que brinda diversas facilidades, su uso es obligatorio en la propuesta solución. En la Figura 4 se puede apreciar la representación en concreto del patrón arquitectónico MVC en la propuesta de solución.



Figura 14: Patrón Modelo-Vista-Controlador en la propuesta solución

Capítulo 2: Análisis y diseño

2.8 Modelo de datos

Las bases de datos son un gran pilar de la programación, ya que permiten almacenar, organizar y manipular cantidades de datos con cierta facilidad. Cuando se utiliza una base de datos para gestionar información, se está plasmando una parte del mundo real en una serie de tablas, registros y campos ubicados en un ordenador; creándose un modelo parcial de la realidad.

En la Figura 15 se representa el modelo físico correspondiente a la base de datos de la propuesta de solución, el mismo fue generado por la herramienta CASE Visual Paradigm.

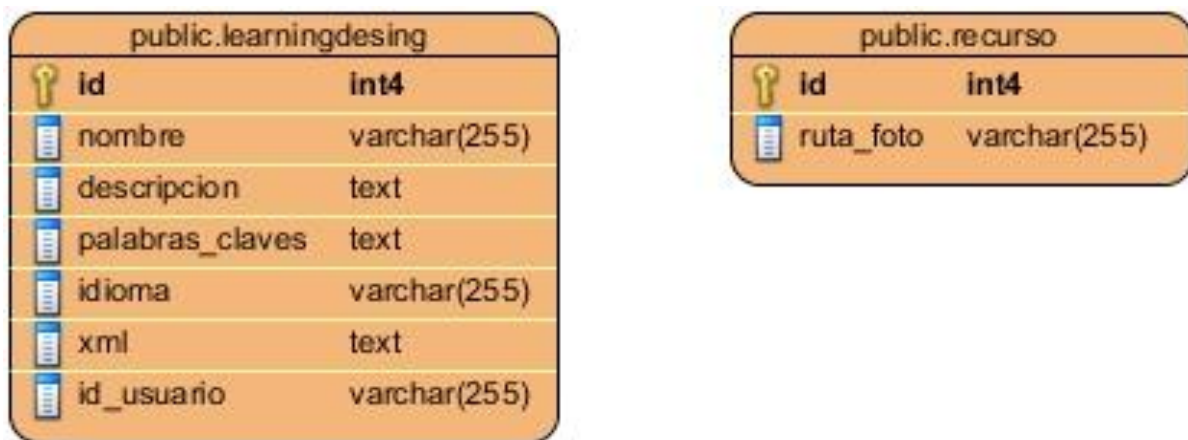


Figura 15: Modelo físico de la base de datos

A continuación se muestra una descripción de las tablas que conforma la base de datos:

Learningdesign: en esta tabla se almacena toda la información de los diseños de aprendizaje creados por el módulo, sus metadatos y contenido en XML, así como un id asignado automáticamente y el id del usuario que lo creó.

Recurso: en esta tabla se almacena la información de los recursos subidos a los diseños de aprendizaje, su id y el nombre del recurso.

2.9 Conclusiones parciales

Luego de realizar el estudio del negocio que abarca la investigación se elaboró la propuesta de solución. Se generaron algunos de los artefactos propuestos por la metodología AUP en su variación para la UCI, como son la descripción y especificación

Capítulo 2: Análisis y diseño

de requisitos, las que permitieron conocer las funcionalidades y características de la propuesta de solución. Se diseñaron los prototipos de interfaz de usuario que constituyen un elemento base para el inicio del proceso de implementación.

Se presentó el diseño del módulo a desarrollar con una propuesta de solución que contiene el patrón arquitectónico Modelo-Vista-Controlador. También se definió el diagrama de clase del diseño mostrando la estructura interna del sistema, fundamentando el uso de patrones de diseño GRASP y GOF para lograr una correcta implementación del módulo.

Capítulo 3: Implementación y pruebas

Capítulo 3: Implementación y pruebas

Introducción

Una vez terminado el diseño, se dispone a abordar los temas relacionados con los flujos de trabajo implementación y prueba. Es por esto que en el presente capítulo se desarrolla una descripción de la implementación. También se especifican las pruebas que se le realizaron al módulo Diseño de aprendizaje, para validar una correcta implementación de los requisitos contribuyendo a elevar la calidad del producto desarrollado.

3.1 Diagrama de componentes

El diagrama de componentes describe la descomposición física del sistema en componentes, a efectos de construcción y funcionamiento. La descomposición del diagrama de componentes se realiza en términos de componentes y de relaciones entre los mismos. Los componentes identifican objetos físicos que hay en tiempo de ejecución, de compilación o desarrollo, y tienen identidad propia y una interfaz bien definida. Cada componente incorpora la implementación de ciertas clases del diseño del sistema. En un diagrama de componentes se muestran las diferentes relaciones de dependencia que se pueden establecer entre componentes.

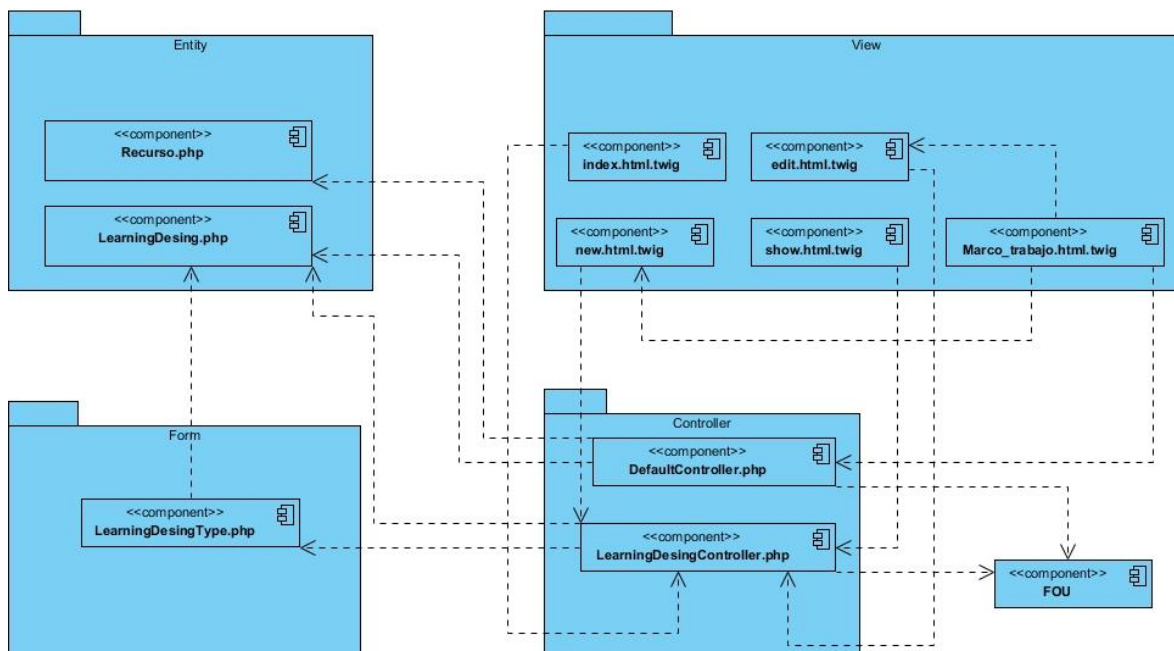


Figura 16: Diagrama de componentes módulo Diseño de aprendizaje

Capítulo 3: Implementación y pruebas

3.2 Implementación del sistema

Como se especifica anteriormente con el resultado del diseño se comienza la implementación, logrando concebir la arquitectura y el sistema como un todo. Además se definen estándares de codificación porque un estilo de programación homogéneo en un proyecto permite que todos los participantes lo puedan entender en menos tiempo y que el código sea accesible para un futuro mantenimiento.

3.2.1 Estándares de codificación

El objetivo fundamental de los estándares de codificación empleados en el módulo Diseño de aprendizaje es lograr una arquitectura y un estilo consistente, con lo cual el sistema resulte fácil de entender y mantener. Tratando de entender el formato y el estilo utilizado en el código escrito por los desarrolladores, buscando aportar una mayor organización y limpieza.

Indentación: las normas de indentación se centran en la lectura y comprensión del código por las personas. El indentado utilizado en el módulo Diseño de aprendizaje debe seguir las siguientes reglas:

- Siempre se indenta 4 espacios el código encerrado entre llaves '{' y '}'.
- El primer corchete está en el mismo renglón que la instrucción.
- A partir de la última llave '}' se termina la indentación.

La figura 17 muestra un ejemplo de la indentación del código presentada a continuación:

```
function contar_informaciones() {
    var tot_info = $('.informacion');
    var nombre;
    var mayor = 0;
    for (i = 0; i < tot_info.length; i++) {
        nombre = tot_info[i].getAttribute('id').replace('objetivo_', '');
        if (nombre > mayor) {
            mayor = nombre;
        }
    }
    var val = parseInt(mayor);
    num_informacion = val;
}
```

Figura 17: Indentación del código

Capítulo 3: Implementación y pruebas

Comentarios: los comentarios son notas cortas explicativas que se agregan al código para aportar mayor información a las personas que lo leen. Los comentarios de línea deben escribirse de forma clara, en el caso de las funcionalidades significativas o clases se usarán comentarios de bloque al inicio de estos. Debe evitarse comentar por cada línea de código. (Ver figura 18)

```
//Seleccionar las informaciones al cliquearlas
$('.informacion').click(function() {
    $('.informacion').css({'background': 'none'});
    $(this).css({'background': '#808080'});
    info_rol = $(this).attr('id');
});
```

Figura 18: Comentarios del código.

Convenciones de nombres: el objetivo fundamental es nombrar las clases y métodos del módulo de forma estándar. En ambos casos se usará el idioma español para hacer más fácil el entendimiento posterior por cualquier desarrollador, en caso de futuros mantenimientos o inclusión de nuevas funcionalidades, si es un nombre compuesto se separaran las palabras usando *underscore* (Ver figura 19)

```
function crear_arbol() {
    $('#mostrar').remove();
    $('#area_arbol').append('<div id="mostrar"><div id="jstree"><ul id="raiz"></ul></div></div>');
    var contenido = "" + $('#raiz_2').html();
    var cont = contenido.split('nodo');
    var palabra = '';
    for (i = 0; i < cont.length; i++) {
        palabra += cont[i] + 'node';
    }
    $('#raiz').append(palabra);
    $('#jstree').jstree();
    jQuery().stop();
}
```

Figura 19: Convenciones de nombre aplicadas en un método

3.2.2 Principales métodos implementados

Para obtener los resultados esperados del módulo Diseño de aprendizaje, fue necesario que se implementaran métodos fundamentales para crear la estructura de un diseño de aprendizaje, así como poder guardar su contenido en la base de datos

Insertar Acto:

Capítulo 3: Implementación y pruebas

```
$('.Obras').click(function(f) {
    lock = 1;
    var offset = $(this).offset();
    var posi_x = f.pageX - offset.left;
    var posi_y = f.pageY - offset.top;
    var nodo = 'hijos_nodo_' + $(this).attr('id');
    if (x == 2 && lock1 == 0) {
        cont_actos += 1;
        $(this).append('<div id="acto' + cont_actos + '" class="actos" style="width: 100px ;
        $('.actos').draggable({revert: "invalid"});
        $('.actos').resizable();
        $('.actos').droppable();
        $('#' + nodo).append('<li id = "nodo_acto' + cont_actos + '">Acto ' + cont_actos + '<
        crear_arbol();
        crear_salva_html();
        asignar_hijos();
        guardar_estructura();
        x = 0;
    }
    lock1 = 0;
});
```

Figura 20: Método insertar acto

Este método es el encargado de insertar un nuevo acto en una obra. Se ejecuta al hacer clic en una obra, se toma la posición del elemento clickeado para poder insertar el nuevo acto en la posición deseada, se comprueba el valor de la variable x, la cual almacena el tipo de elemento que se ha seleccionado y la variable lock que comprueba que solo se esté clickeando una obra, en caso de que las comprobaciones sean válidas se añade el código HTML definido para un acto al elemento de tipo obras clickeado.

Posteriormente le asigna al nuevo acto las propiedades, *draggable*, *resizable* y *droppable*, pertenecientes al *framework* jQuery, las cuales le permitirán al usuario poder mover, cambiar de tamaño y soltar elementos sobre este acto, lo que es fundamental para el desarrollo del módulo, crea el nodo correspondiente al elemento y se lo inserta al nodo correspondiente a la obra, luego se generan los cambios en la vista estructural, se crea una salva del código HTML, se le asigna a la obra que acepte dentro de ella al nuevo acto y se guarda la estructura en la base de datos.

Se tomó como ejemplo este método ya que la inserción de los otros elementos, es similar por lo cual entendiendo el funcionamiento de este método se puede comprender como se insertan todos los elementos.

Guardar información:

Capítulo 3: Implementación y pruebas

```
function guardar_estructura() {
    var texto = $('#html_enviar').val();
    while (texto.contains(' ')) {
        texto = texto.replace(' ', '');
    }
    $('#html_enviar').attr('value', texto);
    var url = "{{path('subir_learning_desing')}}";
    var listado = new Array();
    listado.push({id: $("#id").val(), html: $("#html_enviar").val()});
    $.ajax({
        type: "POST",
        data: {
            name: JSON.stringify(listado)
        },
        url: url,
        datatype: 'json',
        success: function(data) {
            $('#imagen_guardar').fadeIn(600);
            $('#imagen_guardar').fadeOut(600);
        }
    });
}
```

Figura 21: Método guardar_estructuraAction()

El método guardar información es el encargado de enviar los cambios realizados en la estructura del diseño de aprendizaje al método subir_learning_designAction() que se encuentra en la clase php DefaultController, encargado de guardar en la base de datos estas estructuras, este obtiene todo el código del HTML guardado en la variable html_enviar , lo limpia de posibles espacios innecesarios y lo envía utilizando la función ajax , al terminar muestra una imagen como confirmación de que se ha guardado correctamente la información en la base de datos.

Cargar HTML

Capítulo 3: Implementación y pruebas

```
public function cargar_htmlAction() {
    $peticion = $_POST['name'];
    $datos = json_decode($peticion, true);
    $id = $datos[0]['id'];
    $em = $this->getDoctrine()->getManager();
    $desing = $em->getRepository('LearningDesingBundle:LearningDesing')->find($id);
    $html = $desing->getXml();
    $datosHtml = new JsonResponse();
    $datosHtml->setData(array(
        "html" => $html));
    return $datosHtml;
}
```

Figura 22: Método cargar_htmlAction()

Este método pertenece a la clase controladora DefaultController.php el cual permite cargar el código HTML de una estructura guardada en la vista para poder editarla, el cual recibe un identificador, busca el diseño de aprendizaje correspondiente a este id, obtiene su código HTML y lo devuelve a la vista mediante una respuesta Json.

3.3 Validación del módulo Diseño de aprendizaje

La validación es el proceso de evaluación de un sistema para determinar si satisface los requisitos determinados. El desarrollo de un sistema informático implica una serie de actividades evidenciadas durante el desarrollo de la investigación donde pueden aparecer errores desde el inicio del proceso, dentro de los pasos del diseño o en la implementación. Por ello el desarrollo del software debe ir acompañado de una actividad que garantice la calidad del mismo. En estas actividades un sistema es ejecutado bajo condiciones o requerimientos determinados, los resultados son observados, registrados y analizados.

Según el propio Pressman “La prueba del software es un elemento crítico para la garantía de calidad del software y representa una revisión de las especificaciones, del diseño y de la codificación”. Para validar el módulo Diseño de aprendizaje se utilizaron pruebas de caja negra.

3.3.1 Pruebas de caja negra

Son aquellas que se realizan sobre la interfaz del software. Además, no requieren el conocimiento de la estructura interna del programa para su puesta en marcha.

Capítulo 3: Implementación y pruebas

Conociendo la función para la que fue diseñado, se hacen pruebas que demuestren que cada función es operativa y al mismo tiempo se buscan errores en cada una. Permiten obtener un conjunto de condiciones de entrada que ejecuten todos los requisitos funcionales de un programa.

A continuación se muestran tres de los casos de prueba de aceptación realizados a las especificaciones de requisitos expuestas con anterioridad. Los restantes casos de prueba se muestran en el anexo “Casos de prueba de aceptación”.

Tabla 5: CPA Adicionar Diseño de aprendizaje

Caso de prueba de aceptación
Especificación de requisito: Crear Diseño de aprendizaje
Nombre : Crear Diseño de aprendizaje
Descripción: Adiciona un diseño de aprendizaje a la base de datos.
Condiciones de ejecución: El usuario debe estar autenticado.
Pasos de ejecución: <ul style="list-style-type: none">• Ir a la lista de diseños de aprendizaje.• Seleccionar la opción crear diseño de aprendizaje.• Se llenan los datos necesarios.• Se presiona el botón Crear.• Se crea la estructura deseada para el contenido del diseño de aprendizaje.• Se presiona el botón Guardar.
Resultado esperado: El sistema adiciona un curso a la base de datos.
Evaluación de prueba: Prueba satisfactoria.

Tabla 6: CPA Crear acto

Caso de prueba de aceptación
Especificación de requisito: Crear acto.
Nombre : Crear acto

Capítulo 3: Implementación y pruebas

Descripción: Adiciona un acto a una obra en el diseño de aprendizaje.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una obra.
Pasos de ejecución: <ul style="list-style-type: none">• Ir a la lista de diseños de aprendizaje/crear acto de aprendizaje/creación del contenido o acto de aprendizaje/edición del contenido.• Seleccionar la opción método.• Seleccionar la opción acto de aprendizaje.• Dar clic sobre una obra.
Resultado esperado: El sistema adiciona un acto a la obra seleccionada.
Evaluación de prueba: Prueba satisfactoria.

Tabla 7: CPA Editar acto

Caso de prueba de aceptación
Especificación de requisito: Editar acto.
Nombre: Editar acto.
Descripción: Edita acto.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el acto ha sido creado.
Pasos de ejecución: <ul style="list-style-type: none">• Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido.• Se da clic derecho sobre un acto.• Se selecciona la opción editar información.• Se llenan los campos correspondientes.• Se selecciona la opción Aceptar

Capítulo 3: Implementación y pruebas

Resultado esperado: El sistema edita la información de un acto.
Evaluación de prueba: Prueba satisfactoria.

Las pruebas a las funcionalidades se realizaron en 3 iteraciones obteniendo para la 1ra 17 no conformidades significativas, 20 no significativas y 6 recomendaciones. Para la 2da iteración, 6 no conformidades significativas, 9 no significativas y 3 recomendaciones y para la 3ra iteración, no se encontraron no conformidades ni recomendaciones. Las no conformidades no significativas, se centraron en errores ortográficos como: omisiones de tildes, paréntesis y cambio de mayúscula por minúscula y las significativas, en errores de validación y cambios en el diseño.

3.4 Conclusiones parciales

En este capítulo se sentaron las bases para la implementación mostrando el diagrama de componentes perteneciente al módulo Diseño de aprendizaje, para facilitar la implementación del módulo. Se definió un estándar de codificación adecuado para mejorar el entendimiento y claridad del código y se explicaron los principales métodos implementados.

Los métodos de pruebas de caja negra aplicados permitieron validar el correcto funcionamiento del módulo, en la interacción con el usuario, comprobándose que las funcionalidades son las esperadas por el cliente.

Conclusiones generales

Conclusiones generales

Como resultado de la investigación se desarrolló la implementación de un módulo para la creación de diseños de aprendizaje basándose en la especificación IMS-LD para CRODA 3.0, arribando a las siguientes conclusiones:

- La investigación realizada sobre los diseños de aprendizaje basados en la especificación IMS-LD, permitieron constatar que los mismos son de vital importancia para el desarrollo del aprendizaje electrónico puesto que facilita la elaboración del material por parte de los involucrados en la producción, la gestión del proceso a los profesores y la ejecución del mismo por los estudiantes.
- Se desarrolló un módulo para la creación de diseños de aprendizaje basados en la especificación IMS-LD para la herramienta de autor CRODA 3.0 que se caracteriza por ser muy intuitivo, de diseño atractivo y con gran usabilidad.
- Se probaron de forma satisfactoria las funcionalidades desarrolladas en el módulo a través de pruebas de Caja Negra, detectándose en este proceso algunas no conformidades que fueron corregidas para lograr un producto final de mayor calidad.

Recomendaciones

Recomendaciones

- Incluir en el módulo Diseño de aprendizaje para CRODA 3.0 el nivel B y C de la especificación IMS-LD.

Referencia bibliográfica

Referencias bibliográficas

Berlanga, A. J. (2005) IMS Learning Design: Hacia la Descripción Estandarizada de los Procesos de Enseñanza.

Griffiths, D., Blat, J., García, R., & Sayago, S. (2005). La aportación de IMS Learning Design a la creación de recursos pedagógicos reutilizables. Revista de Educación a Distancia.

IMS (2015). "IMS Learning Design Information Model". disponible en: http://www.imsglobal.org/learningdesign/ldv1p0/imslid_infov1p0.html

IMS (2015). "IMS GLC: About IMS". Disponible en: <http://www.imsglobal.org/aboutims.html>.

O'FARRILL, José L. Montero; TUNIS, Elsa Herrero. Las herramientas de autor en el proceso de producción de cursos en formato digital. Pixel-Bit. Revista de Medios y Educación, (33), 2008, p. 59-72.

Apache.org (2013). "Welcome to the Apache Software Foundation!" disponible en: <http://apache.org/>

Falcon, M. M. (2011) Implementación del nivel A de IMS-LD para la herramienta de autor web CRODA.

Cuaresma, S. B. (2008). "Metodologías de desarrollo." disponible en: <http://www.marblestation.com/?p=644>.

Jacobson, I. and G. Booch (2000). El proceso unificado de desarrollo de software.

Consumoteca (2009). "Servidor Web. Consumoteca." disponible en: <http://www.consumoteca.com/diccionario/servidor-web>.

Navathe, S. B. (1997) Fundamentos de Sistemas de Bases de Datos.

Quiñones, E. (2008) Introducción a postgresql.

Zaninotto, F. P. F. (2007). The Definitive Guide to symfony.

Referencia bibliográfica

Domínguez-Dorado, M. (2005). Todo Programación. Nº 13.

Koper, R. and C. Tattersall (2005) Diseño de aprendizaje: un manual sobre el modelado y la entrega de la educación y la formación en red.

Bootstrap (2015). Introducing Bootstrap disponible en: <http://getbootstrap.com/2.3.2/>.

Visual Paradigm (2015). Features List disponible en: <http://www.visual-paradigm.com/features/>.

Pressman, Roger S. (2002). Ingeniería de software: Un enfoque práctico 5ta edición. s.l. : McGraw Hill. pág. 238.

Y. D. González y Y. F. Romero (2012). Patrón Modelo-Vista-Controlador. Revista Telem@tica, vol. 11, no 1.

Catalani, Exequiel.: “Arquitectura Modelo/Vista/Controlador”, disponible en:

<http://exequielc.wordpress.com/2007/08/20/arquitectura-modelovistacontrolador/>.

Olivares Rojas, Juan Carlos (2007). Patrones de diseño. Sistema Nacional de Educación Superior Tecnológica : s.n.

Larman, Craig (1999). UML y Patrones. Introducción al análisis orientado a objetos. págs. 85-88.

Erich Gamma, Richard Helm (1995). Design Patterns: Elements of Reusable Object-Oriented Software.

Mühlrad, Daniel (2008). Patrones de diseño.

Lenguajes de Programación (2009). Disponible en : <http://www.lenguajes-de-programacion.com>.

Instituto de tecnologías educativas (2015). “Estándares en e-learning y diseño educativo”. Disponible en: <http://ares.cnice.mec.es/informes/20/index.htm>.

Referencia bibliográfica

CRUZ CASTRO, Julio, 2011, Sistema de Gestión de Calidad del Grupo de Calidad de FORTES. Módulo de Capacitación. Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas. La Habana, Cuba: Universidad de las Ciencias Informáticas. [Accedido 11 Marzo 2015]. Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_04629_11/1/TD_04629_11.pdf/.

e-ABC. 2010. Disponible en: <http://www.e-abclearning.com/definicion-e-learning>.

Burgos, D. (2005) IMS Learning Design desde dentro. Una especificación para crear escenarios de aprendizaje online (I Entrega).

Pressman, Roger S. Ingeniería de Software: Un enfoque práctico. 6a. ed. s.l. : McGraw-Hill. pp 226,420.

Iznaola Cuscó, Marta Bárbara. 2008. "La satisfacción del profesor de Educación Física".n. 27-2, 27-35, s.l. : Funámbulos Editores.

Rodríguez, D. 2002. "La técnica de ladov. Una aplicación para el estudio de la satisfacción de los alumnos para la clase de educación física". Revista Digital-Buenos Aires.

Centro de formación permanente de la Universidad de Sevilla. 2015. Disponible en <http://www.cfp.us.es/e-learning-definicion-y-caracteristicas>.

Manolo Torres. 2015. "Information Systems Group University of Almeria". Disponible en <http://indalog.ual.es/mtorres/LP/Prueba.pdf>.

Tablas de especificación de requisitos:

Tabla 10: Especificación de requisito Editar diseño de aprendizaje.

Precondiciones	El usuario debe estar autenticado y el diseño de aprendizaje ha sido creado.
Flujo de eventos	
Flujo básico: Editar diseño de aprendizaje.	
1	El sistema muestra la lista de los diseños de aprendizaje existentes.
2	Se selecciona el botón Editar presente en uno de los diseños de aprendizaje.
3	Se muestra un formulario con los siguientes campos con los datos de un diseño de aprendizaje editables: <ul style="list-style-type: none"> • Nombre • Descripción • Palabras Claves • Idioma
4	Se editan los datos.
5	Se selecciona la opción editar diseño de aprendizaje.
6	El sistema valida los campos insertados
7	Si los datos son correctos el sistema edita el diseño de aprendizaje.
8	Se muestra la vista de creación del contenido del diseño de aprendizaje.
9	El sistema carga la estructura del diseño de aprendizaje.
10	Se edita la estructura del contenido del diseño de aprendizaje
11	Se presiona el botón guardar.
12	Concluye el requisito
Post-condiciones	
1.	Se edita un diseño de aprendizaje.
Flujos alternativos	
Flujo alternativo 6.a Datos incorrectos	

4	El sistema señala los datos incorrectos y permite corregirlos.	
5	Se corrigen los datos.	
6	Volver al paso 5 del flujo básico.	
Post-condiciones		
2	N/A	
Flujo alternativo 5.a El usuario cancela la acción.		
1	El sistema muestra la lista de diseños de aprendizaje	
2	Concluye el requisito.	
Post-condiciones		
1	No se edita el diseño de aprendizaje.	
2	Se muestra la lista de diseños de aprendizaje disponibles.	
Flujo alternativo 11.a El usuario cancela la acción.		
1	El sistema muestra la lista de diseños de aprendizaje	
2	Concluye el requisito.	
Post-condiciones		
1	Se muestra la lista de diseños de aprendizaje disponibles.	
Validaciones		
1.	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Editar diseño de aprendizaje

Nombre

Descripción

Palabras

Idioma

Regresar a la lista Crear

Figura 24: Interfaz gráfica de usuario

Tabla 11: Especificación de requisito Eliminar diseño de aprendizaje.

Precondiciones	El Diseño de aprendizaje ha sido creado y el usuario está autenticado.
Flujo de eventos	
Flujo básico: Eliminar Diseño de aprendizaje.	
1.	El sistema muestra la lista de los diseños de aprendizaje existentes.
2.	Se selecciona el botón eliminar.
3.	Se elimina el Patrón.

4.	Concluye el requisito.	
Post-condiciones		
1.	Se elimina el Patrón.	
Flujos alternativos		
-		
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	



Figura 25: Interfaz gráfica de usuario

Tabla 12: Especificación de requisito Crear objetivos generales.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado y se encuentra en la vista
-----------------------	--

de creación del contenido del diseño de aprendizaje.	
Flujo de eventos	
Flujo básico: Crear Objetivos generales.	
1	Se selecciona la opción objetivos.
2	Se selecciona el botón agregar nuevo objetivo.
3	Se muestra un formulario con los siguientes campos a llenar: <ul style="list-style-type: none"> • Nombre • Referencia
4	Se selecciona la opción Aceptar.
5	El sistema valida los campos insertados
6	Si los datos son correctos el sistema agrega un nuevo objetivo.
7	Concluye el requisito
Post-condiciones	
2.	Se crea un objetivo general.
Flujos alternativos	
Flujo alternativo 5.a Datos incorrectos	
1	El sistema señala los datos incorrectos y permite corregirlos.
2	Se corrigen los datos.
3	Volver al paso 4 del flujo básico.
Post-condiciones	
1	N/A
Flujo alternativo 4.a El usuario cancela la acción	
1	Concluye el requisito.
Post-condiciones	
1	No se crea el objetivo general.
Validaciones	
2.	Se validan los datos.

Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

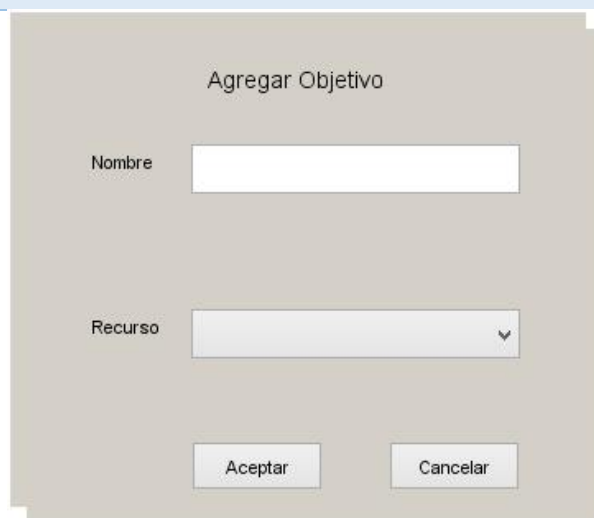


Figura 26: Interfaz gráfica de usuario

Tabla 13: Especificación de requisito Editar objetivo general.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el objetivo ha sido creado.
Flujo de eventos	
Flujo básico: Editar objetivo general del diseño de aprendizaje.	
1	Se selecciona la opción objetivos.
2	Se selecciona un objetivo general.
3	Se selecciona el botón editar objetivo.

4	Se muestra un formulario con los siguientes campos con los datos del objetivo general: <ul style="list-style-type: none"> • Nombre • Referencia 	
5	Se editan los datos	
6	Se selecciona la opción Aceptar.	
7	El sistema valida los campos insertados	
8	Si los datos son correctos el sistema edita el objetivo general.	
9	Concluye el requisito	
Post-condiciones		
3.	Se edita un objetivo general.	
Flujos alternativos		
Flujo alternativo 7.a Datos incorrectos		
1	El sistema señala los datos incorrectos y permite corregirlos.	
2	Se corrigen los datos.	
3	Volver al paso 6 del flujo básico.	
Post-condiciones		
1	N/A	
Flujo alternativo 6.a El usuario cancela la acción		
1	Concluye el requisito.	
Post-condiciones		
1	No se edita el objetivo general.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	

Asuntos pendientes	N/A
---------------------------	-----

Figura 27: Interfaz gráfica de usuario

Tabla 14: Especificación de requisito Eliminar objetivo general.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el objetivo ha sido creado.
Flujo de eventos	
Flujo básico: Eliminar objetivo general.	
1.	Se selecciona la opción objetivos.
2.	Se selecciona el objetivo a eliminar.
3	Se selecciona la opción eliminar.
4	Se elimina el objetivo general.
5	Concluye el requisito.

Post-condiciones		
1	Se elimina el objetivo general.	
Flujos alternativos		
-		
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	



Figura 28: Interfaz gráfica de usuario

Tabla 15: Especificación de requisito Crear prerrequisito.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado y se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Flujo de eventos	
Flujo básico: Crear prerrequisito.	

1	Se selecciona la opción prerequisites.	
2	Se selecciona el botón agregar nuevo prerequisite.	
3	Se muestra un formulario con los siguientes campos a llenar: <ul style="list-style-type: none"> Nombre Referencia 	
4	Se selecciona la opción Aceptar.	
5	El sistema valida los campos insertados	
6	Si los datos son correctos el sistema agrega un nuevo prerequisite.	
7	Concluye el requisito	
Post-condiciones		
4.	Se crea un prerequisite.	
Flujos alternativos		
Flujo alternativo 5.a Datos incorrectos		
1	El sistema señala los datos incorrectos y permite corregirlos.	
2	Se corrigen los datos.	
3	Volver al paso 4 del flujo básico.	
Post-condiciones		
1	N/A	
Flujo alternativo 4.a El usuario cancela la acción		
1	Concluye el requisito.	
Post-condiciones		
1	No se crea el prerequisite.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos	N/A	

especiales	
Asuntos pendientes	N/A

Figura 29: Interfaz gráfica de usuario

Tabla 16: Especificación de requisito Editar prerrequisito.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el prerrequisito ha sido creado.
Flujo de eventos	
Flujo básico: Editar prerrequisito.	
1	Se selecciona la opción prerrequisitos.
2	Se selecciona un prerrequisito.
3	Se selecciona el botón editar prerrequisito.
4	Se muestra un formulario con los siguientes campos con los datos del prerrequisito:

	<ul style="list-style-type: none"> Nombre Referencia 				
5	Se editan los datos				
6	Se selecciona la opción Aceptar.				
7	El sistema valida los campos insertados				
8	Si los datos son correctos el sistema edita el prerequisite.				
9	Concluye el requisito.				
Post-condiciones					
1	Se edita un prerequisite.				
Flujos alternativos					
Flujo alternativo 7.a Datos incorrectos					
1	El sistema señala los datos incorrectos y permite corregirlos.				
2	Se corrigen los datos.				
3	Volver al paso 5 del flujo básico.				
Post-condiciones					
1	N/A				
Flujo alternativo 6.a El usuario cancela la acción					
1	Concluye el requisito.				
Post-condiciones					
1	No se edita el prerequisite.				
Validaciones					
1	Se validan los datos.				
Conceptos	Usuario <table border="1" style="margin-left: 20px;"> <tr> <td colspan="2">Atributos del concepto que se utilizan en el requisito:</td> </tr> <tr> <td colspan="2"> <ul style="list-style-type: none"> Visibles en la interfaz: Nombre de usuario </td> </tr> </table>	Atributos del concepto que se utilizan en el requisito:		<ul style="list-style-type: none"> Visibles en la interfaz: Nombre de usuario	
Atributos del concepto que se utilizan en el requisito:					
<ul style="list-style-type: none"> Visibles en la interfaz: Nombre de usuario					
Requisitos especiales	N/A				
Asuntos	N/A				

pendientes	
<p>Editar Prerrequisito</p> <p>Nombre <input type="text"/></p> <p>Recurso <input type="text" value="v"/></p> <p style="text-align: center;"> <input type="button" value="Aceptar"/> <input type="button" value="Cancelar"/> </p>	

Figura 30: Interfaz gráfica de usuario

Tabla 17: Especificación de requisito Eliminar prerrequisito.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el prerrequisito ha sido creado.
Flujo de eventos	
Flujo básico: Eliminar prerrequisito.	
1.	Se selecciona la opción prerrequisitos.
2.	Se selecciona el prerrequisito a eliminar.
3	Se selecciona la opción eliminar.
4	Se elimina el prerrequisito.
5	Concluye el requisito.
Post-condiciones	

1	Se elimina el prerrequisito.	
Flujos alternativos		
-		
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

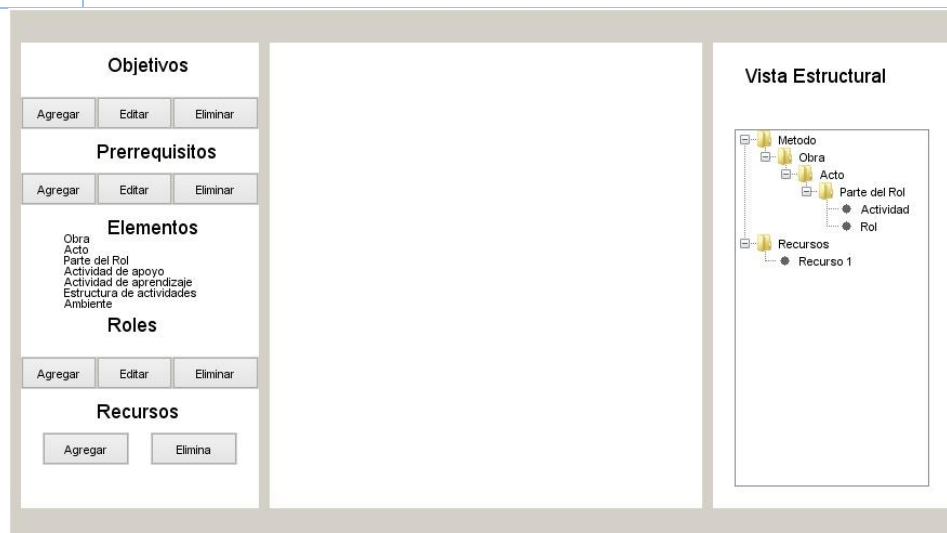


Figura 31: Interfaz gráfica de usuario

Tabla 18: Especificación de requisito Crear rol.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado y se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Flujo de eventos	
Flujo básico: Crear Rol.	

1	Se selecciona la opción roles.
2	Se selecciona el botón agregar nuevo rol.
3	Se muestra un formulario con los siguientes campos a llenar: <ul style="list-style-type: none"> • Nombre • Propiedad creador • Responsabilidad • Referencia en la web • Máximo en el rol • Mínimo en el rol
4	Se selecciona el tipo de rol.
5	Se selecciona la opción Agregar.
6	El sistema valida los campos insertados
7	Si los datos son correctos el sistema agrega un nuevo rol.
8	Concluye el requisito
Post-condiciones	
1	Se crea un rol.
Flujos alternativos	
Flujo alternativo 6.a Datos incorrectos	
1	El sistema señala los datos incorrectos y permite corregirlos.
2	Se corrigen los datos.
3	Volver al paso 5 del flujo básico.
Post-condiciones	
1	N/A
Flujo alternativo 5.a El usuario cancela la acción	
1	Concluye el requisito.
Post-condiciones	
1	No se crea rol.

Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Figura 32: Interfaz gráfica de usuario

Tabla 19: Especificación de requisito Editar rol.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el rol ha sido creado.
-----------------------	--

Flujo de eventos	
Flujo básico: Editar rol.	
1	Se selecciona la opción roles.
2	Se selecciona un rol.
3	Se selecciona el botón editar rol.
4	Se muestra un formulario con los siguientes campos con los datos del rol: <ul style="list-style-type: none"> • Nombre • Tipo de rol • Propiedad creador • Responsabilidad • Referencia web • Máximo en el rol • Mínimo en el rol
5	Se editan los datos
6	Se selecciona la opción Aceptar.
7	El sistema valida los campos insertados
8	Si los datos son correctos el sistema edita el rol.
9	Concluye el requisito
Post-condiciones	
1	Se edita un rol.
Flujos alternativos	
Flujo alternativo 7.a Datos incorrectos	
1	El sistema señala los datos incorrectos y permite corregirlos.
2	Se corrigen los datos.
3	Volver al paso 5 del flujo básico.
Post-condiciones	
1	N/A

Flujo alternativo 6.a El usuario cancela la acción		
1	Concluye el requisito.	
Post-condiciones		
1	No se edita el rol.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Editar Roles

Nombre

Tipo de rol

Profesor Aprendiz

Creador Responsabilidad

Referencia en la web :

Información de rol

Figura 33: Interfaz gráfica de usuario

Tabla 19: Especificación de requisito Eliminar rol.

Precondiciones		El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el rol ha sido creado.
Flujo de eventos		
Flujo básico: Eliminar rol.		
1.	Se selecciona la opción roles.	
2.	Se selecciona el rol a eliminar.	
3	Se selecciona la opción eliminar.	
4	Se elimina el rol.	
5	Concluye el requisito.	
Post-condiciones		
1	Se elimina el rol.	
Flujos alternativos		
-		
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	



Figura 34: Interfaz gráfica de usuario

Tabla 20: Especificación de requisito Crear ambiente.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una parte del rol.
Flujo de eventos	
Flujo básico: Crear ambiente.	
1	Se selecciona la opción método.
2	Se selecciona el elemento ambiente
3	Se da un clic en una parte del rol creada.
4	El sistema agrega el ambiente a la parte del rol cliqueada.
Post-condiciones	
1	Se Agrega el ambiente a la parte del rol.
Flujos alternativos	

Flujo alternativo 3.a La parte del rol tiene un ambiente, una actividad de aprendizaje, una actividad de apoyo o una estructura de actividades insertada.		
1	Concluye el requisito	
Post-condiciones		
1	No se agrega el ambiente	
Validaciones		
1	Se valida que la parte del rol no tenga un ambiente, una actividad de aprendizaje, una actividad de apoyo o una estructura de actividades insertada.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

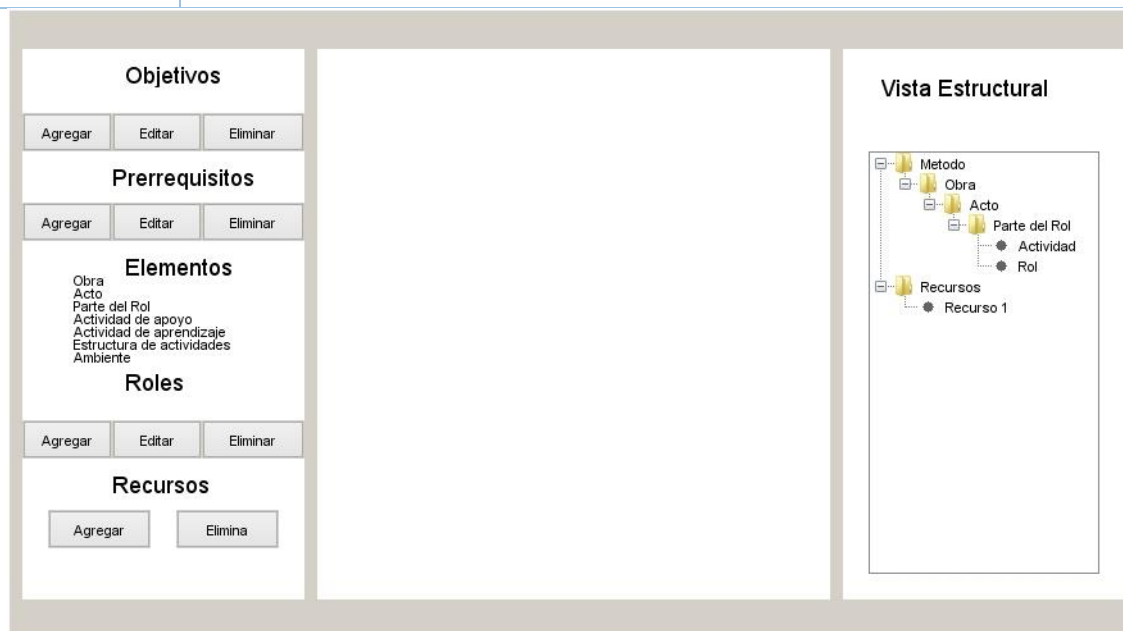


Figura 35: Interfaz gráfica de usuario

Tabla 21: Especificación de requisito Editar ambiente.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el ambiente ha sido creado.
Flujo de eventos	
Flujo básico: Editar ambiente.	
1	Se da clic derecho sobre el ambiente a editar.
2	Se selecciona la opción editar información.
3	Se muestra un formulario con los siguientes campos con los datos del ambiente: <ul style="list-style-type: none">• Nombre del objeto de aprendizaje• Tipo, que puede ser: objeto de conocimiento, objeto de herramienta, objeto test y otros• Una lista con los recursos asociados al objeto de aprendizaje• Nombre de la conferencia• Administrador, que debe ser un rol creado previamente.• Moderador que debe ser un rol creado previamente.• Tipo que puede ser asíncrono, síncrono o anuncio• Una lista con los roles participantes• Una lista con los roles observadores• Una lista con los recursos asociados a la conferencia• Nombre del correo electrónico• Tipo que puede ser personal en el rol o todos en el rol• Rol, que debe ser un rol creado previamente.• Nombre del índice

	<ul style="list-style-type: none"> • Tipo de elemento, que puede ser objeto de aprendizaje, conferencia, correo electrónico, actividad o estructura de actividades. • Elemento, que debe contener los elementos que cumplan con el tipo de elementos seleccionado anteriormente. • Clase del índice. • Tipo de búsqueda, que puede ser Texto libre de búsqueda, índice con referencia e índice sin referencia. • Nombre del ambiente a referenciar
4	Se editan los datos
5	Se selecciona la opción Aceptar.
6	El sistema valida los campos insertados
7	Si los datos son correctos el sistema edita el ambiente.
8	Concluye el requisito
Post-condiciones	
9	Se edita un ambiente.
Flujos alternativos	
Flujo alternativo 6.a Datos incorrectos	
1	El sistema señala los datos incorrectos y permite corregirlos.
2	Se corrigen los datos.
3	Volver al paso 5 del flujo básico.
Post-condiciones	
1	N/A
Flujo alternativo 7.a El usuario cancela la acción	
1	Concluye el requisito.
Post-condiciones	

1	No se edita el ambiente.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none">• Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Editar Informacion ambiente

Nombre del objeto

Tipo del objeto

Recursos del objeto

Nombre conferencia

Administrador conferencia

Moderador conferencia

Tipo conferencia

participantes conferencia observadores conferencia Recursos conferencia

Nombre correo

Tipo Correo

Rol

Nombre indice

Tipo Elemento

Elemento

Clase del indice

Tipo Busqueda

Referencia

Figura 36: Interfaz gráfica de usuario

Tabla 22: Especificación de requisito Eliminar ambiente.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el ambiente ha sido creado.	
Flujo de eventos		
Flujo básico: Eliminar ambiente.		
1.	Se da clic derecho sobre un ambiente.	
2.	Se selecciona la opción eliminar.	
3	El sistema muestra un mensaje pidiendo la confirmación de eliminar.	
4	Se acepta la confirmación.	
5	El sistema elimina el ambiente.	
6	Concluye el requisito	
Post-condiciones		
1	Se elimina el ambiente.	
Flujo alternativo 4.a El usuario cancela la acción		
1	Concluye el requisito	
Post-condiciones		
1	No se elimina el ambiente	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos	N/A	

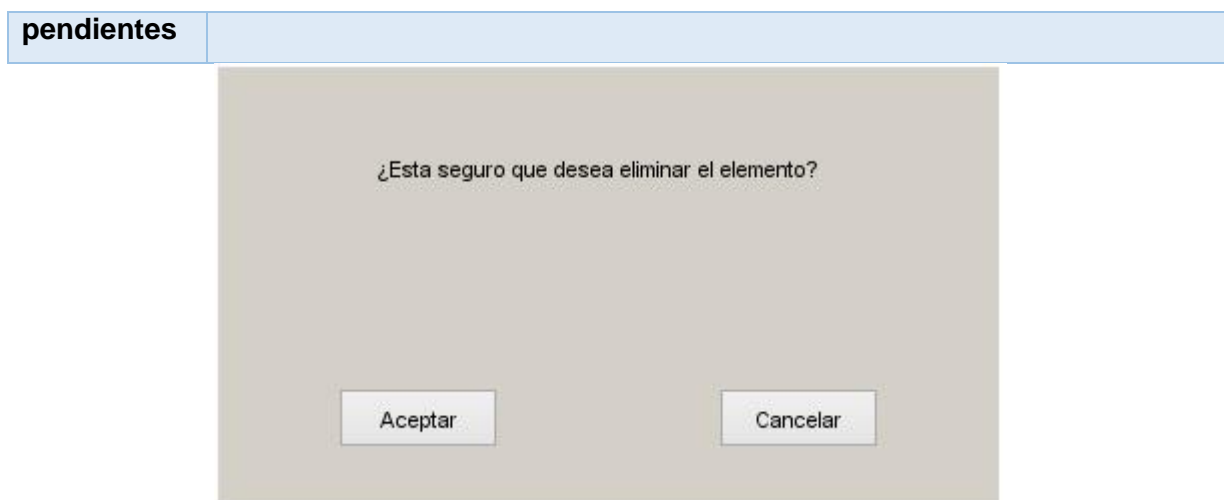


Figura 37: Interfaz gráfica de usuario

Tabla 23: Especificación de requisito Crear actividad de aprendizaje.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una parte del rol.
Flujo de eventos	
Flujo básico: Crear actividad de aprendizaje.	
1	Se selecciona la opción método.
2	Se selecciona el elemento actividad de aprendizaje
3	Se da un clic en una parte del rol creada.
4	El sistema agrega la actividad de aprendizaje a la parte del rol cliqueada.
Post-condiciones	
1	Se Agrega la actividad de aprendizaje a la parte del rol.
Flujos alternativos	
Flujo alternativo 3.a La parte del rol tiene un ambiente, una actividad de aprendizaje, una actividad de apoyo o una estructura de actividades insertada.	

1	Concluye el requisito	
Post-condiciones		
1	No se agrega la actividad de aprendizaje.	
Validaciones		
1	Se valida que la parte del rol no tenga un ambiente, una actividad de aprendizaje, una actividad de apoyo o una estructura de actividades agregado.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

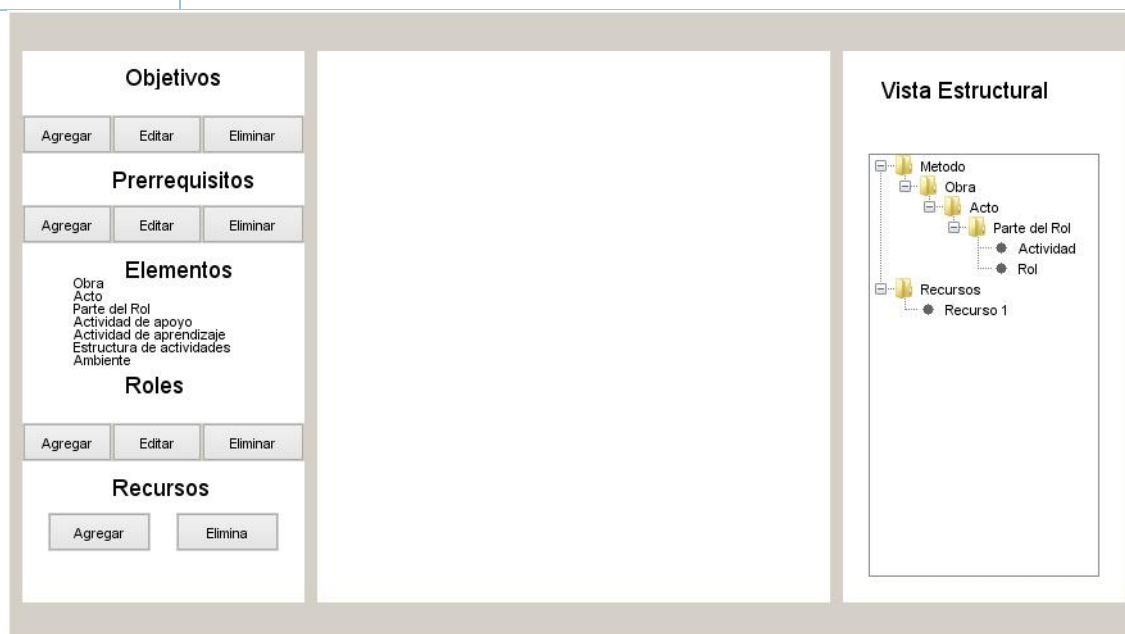


Figura 38: Interfaz gráfica de usuario

Tabla 24: Especificación de requisito Editar actividad de aprendizaje.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de
-----------------------	--

	creación del contenido del diseño de aprendizaje y la actividad de aprendizaje ha sido creada.
Flujo de eventos	
Flujo básico: Editar actividad de aprendizaje.	
1	Se da clic derecho sobre la actividad de aprendizaje a editar.
2	Se selecciona la opción editar información.
3	Se muestra un formulario con los siguientes campos con los datos de la actividad: <ul style="list-style-type: none"> • Descripción de la actividad. • Objetivos de la actividad • Prerrequisitos de la actividad • Completamiento de la actividad Retroalimentación de la actividad
4	Se editan los datos
5	Se selecciona la opción Aceptar.
6	El sistema valida los campos insertados
7	Si los datos son correctos el sistema edita la actividad de aprendizaje.
8	Concluye el requisito
Post-condiciones	
9	Se edita una actividad de aprendizaje.
Flujos alternativos	
Flujo alternativo 6.a Datos incorrectos	
1	El sistema señala los datos incorrectos y permite corregirlos.
2	Se corrigen los datos.
3	Volver al paso 5 del flujo básico.
Post-condiciones	

1	N/A	
Flujo alternativo 5.a El usuario cancela la acción		
1	Concluye el requisito.	
Post-condiciones		
1	No se edita la actividad de aprendizaje.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Editar Información de Actividad de aprendizaje

Descripción

Agregar Eliminar

Objetivos

Agregar Editar Eliminar

Prerrequisitos

Agregar Editar Eliminar

Agregar Editar Eliminar

Completar Actividad

Escogido por el usuario Por tiempo

Años Dias Minutos
Meses Horas Segundos

Retroalimentación

Aceptar Cancelar

Figura 39: Interfaz gráfica de usuario

Tabla 25: Especificación de requisito Eliminar actividad de aprendizaje.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y la actividad de aprendizaje ha sido creada.	
Flujo de eventos		
Flujo básico: Eliminar actividad de aprendizaje.		
1.	Se da clic derecho sobre una actividad de aprendizaje.	
2.	Se selecciona la opción eliminar.	
3	El sistema muestra un mensaje pidiendo la confirmación de eliminar.	
4	Se acepta la confirmación.	
5	El sistema elimina la actividad de aprendizaje.	
6	Concluye el requisito	
Post-condiciones		
1	Se elimina la actividad de aprendizaje.	
Flujo alternativo 4.a El usuario cancela la acción		
1	Concluye el requisito	
Post-condiciones		
1	No se elimina la actividad de aprendizaje	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

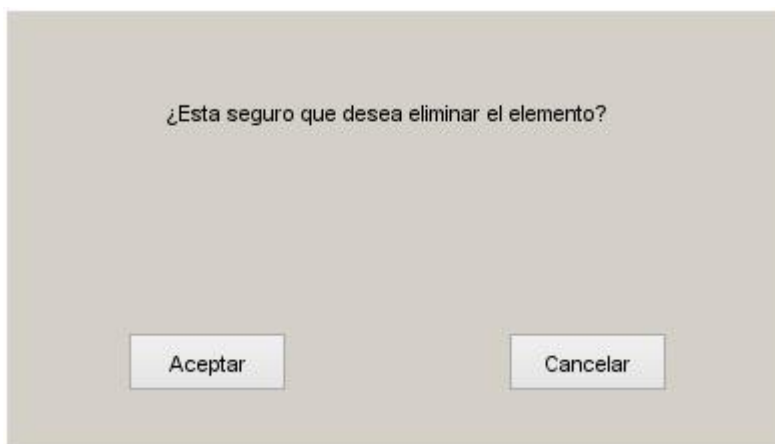


Figura 40: Interfaz gráfica de usuario

Tabla 26: Especificación de requisito Crear actividad de apoyo.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una parte del rol.
Flujo de eventos	
Flujo básico: Crear actividad de apoyo.	
1	Se selecciona la opción método.
2	Se selecciona el elemento actividad de apoyo
3	Se da un clic en una parte del rol creada.
4	El sistema agrega la actividad de apoyo a la parte del rol cliqueada.
Post-condiciones	
1	Se Agrega la actividad de apoyo a la parte del rol.
Flujos alternativos	
Flujo alternativo 3.a La parte del rol tiene un ambiente, una actividad de aprendizaje, una actividad de apoyo o una estructura de actividades insertada.	
1	Concluye el requisito

Post-condiciones		
1	No se agrega la actividad de apoyo.	
Validaciones		
1	Se valida que la parte del rol no tenga un ambiente, una actividad de aprendizaje, una actividad de apoyo o una estructura de actividades agregado.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

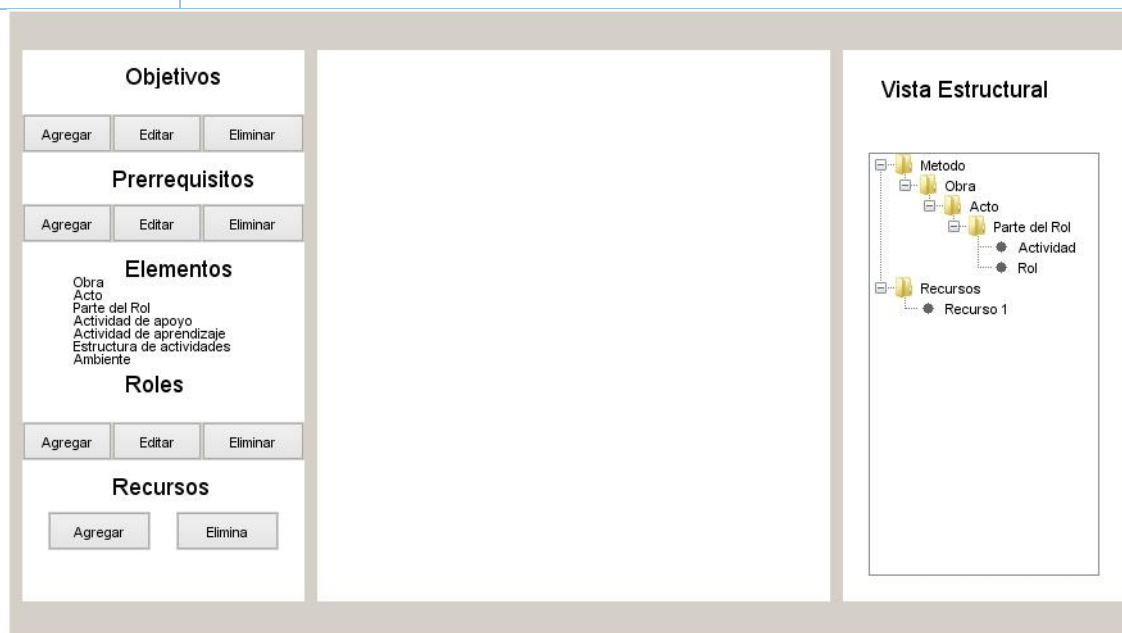


Figura 41: Interfaz gráfica de usuario

Tabla 27: Especificación de requisito Editar actividad de apoyo.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y
-----------------------	---

	la actividad de apoyo ha sido creada.
Flujo de eventos	
Flujo básico: Editar actividad de apoyo del diseño de aprendizaje.	
1	Se da clic derecho sobre la actividad de apoyo a editar.
2	Se selecciona la opción editar información.
3	Se muestra un formulario con los siguientes campos con los datos de la actividad: <ul style="list-style-type: none"> • Descripción de la actividad. • Completamiento de la actividad • Retroalimentación de la actividad
4	Se editan los datos
5	Se selecciona la opción Aceptar.
6	El sistema valida los campos insertados
7	Si los datos son correctos el sistema edita la actividad de apoyo.
8	Concluye el requisito
Post-condiciones	
1	Se edita una actividad de apoyo.
Flujos alternativos	
Flujo alternativo 6.a Datos incorrectos	
1	El sistema señala los datos incorrectos y permite corregirlos.
2	Se corrigen los datos.
3	Volver al paso 5 del flujo básico.
Post-condiciones	
1	N/A
Flujo alternativo 5.a El usuario cancela la acción	
1	Concluye el requisito.
Post-condiciones	

1	No se edita la actividad de apoyo.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Editar Información de Actividad de Apoyo

Descripción

Completar Actividad

Escogido por el usuario
 Por tiempo

Años
 Días
 Minutos
 Meses
 Horas
 Segundos

Retroalimentación

Figura 42: Interfaz gráfica de usuario

Tabla 27: Especificación de requisito Eliminar actividad de apoyo.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y la actividad de apoyo ha sido creada.	
Flujo de eventos		
Flujo básico: Eliminar actividad de apoyo.		
1.	Se da clic derecho sobre una actividad de apoyo.	
2.	Se selecciona la opción eliminar.	
3	El sistema muestra un mensaje pidiendo la confirmación de eliminar.	
4	Se acepta la confirmación.	
5	El sistema elimina la actividad de apoyo	
6	Concluye el requisito	
Post-condiciones		
1	Se elimina la actividad de apoyo.	
Flujo alternativo 4.a El usuario cancela la acción		
1	Concluye el requisito	
Post-condiciones		
1	No se elimina la actividad de apoyo.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

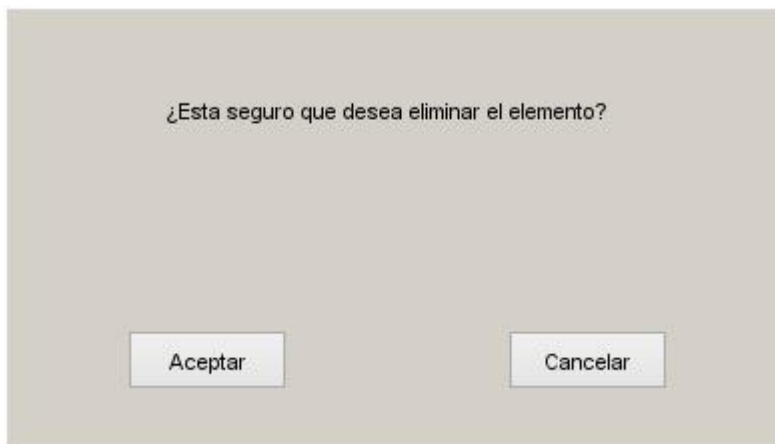


Figura 43: Interfaz gráfica de usuario

Tabla 28: Especificación de requisito Crear estructura de actividades.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una parte del rol.
Flujo de eventos	
Flujo básico: Crear estructura de actividades.	
1	Se selecciona la opción método.
2	Se selecciona el elemento estructura
3	Se da un clic en una parte del rol creada.
4	El sistema agrega la estructura a la parte del rol cliqueada.
Post-condiciones	
1	Se Agrega la estructura a la parte del rol.
Flujos alternativos	
Flujo alternativo 3.a La parte del rol tiene un ambiente, una actividad de aprendizaje, una actividad de apoyo o una estructura de actividades insertada.	
1	Concluye el requisito

Post-condiciones		
1	No se agrega la estructura.	
Validaciones		
1	Se valida que la parte del rol no tenga un ambiente, una actividad de aprendizaje, una actividad de apoyo o una estructura de actividades agregado.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

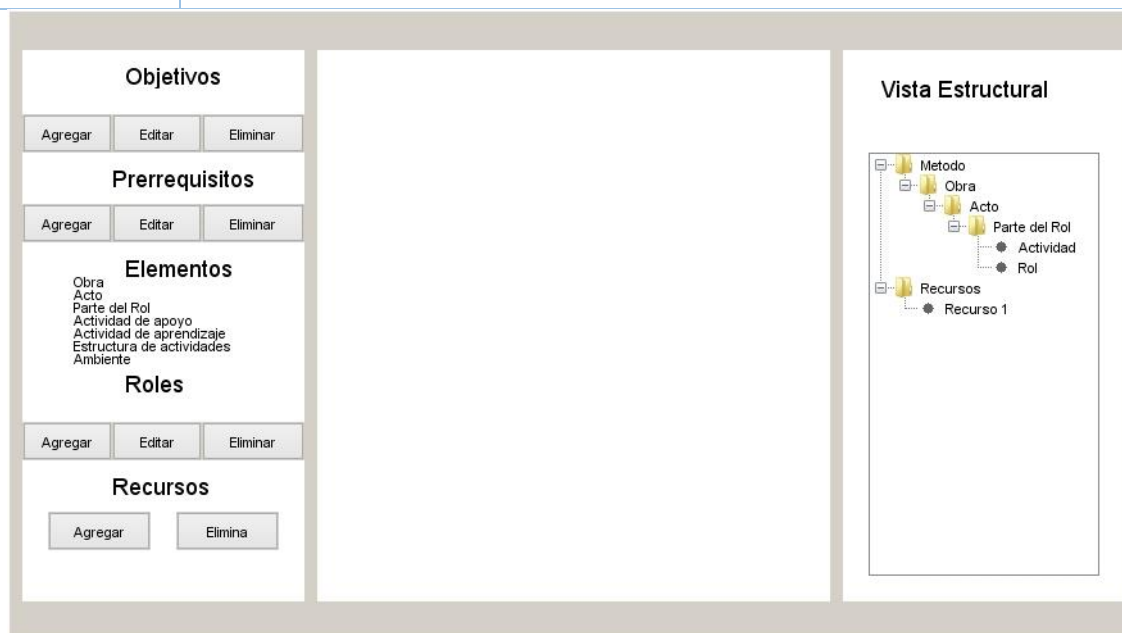


Figura 44: Interfaz gráfica de usuario

Tabla 29: Especificación de requisito Editar estructura.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y
-----------------------	---

	la estructura de actividades ha sido creada.
Flujo de eventos	
Flujo básico: Editar estructura de actividades.	
1	Se da clic derecho sobre la estructura a editar.
2	Se selecciona la opción renombrar elemento.
3	Se muestra un formulario con los siguientes campos con los datos de la actividad: <ul style="list-style-type: none"> • Nombre
4	Se editan los datos
5	Se selecciona la opción Aceptar.
6	El sistema valida los campos insertados
7	Si los datos son correctos el sistema edita la estructura.
8	Concluye el requisito
Post-condiciones	
1	Se edita una estructura de actividades.
Flujos alternativos	
Flujo alternativo 6.a Datos incorrectos	
1	El sistema señala los datos incorrectos y permite corregirlos.
2	Se corrigen los datos.
3	Volver al paso 5 del flujo básico.
Post-condiciones	
1	N/A
Flujo alternativo 5.a El usuario cancela la acción	
1	Concluye el requisito.
Post-condiciones	
1	No se edita la estructura.
Validaciones	
1	Se validan los datos.

Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

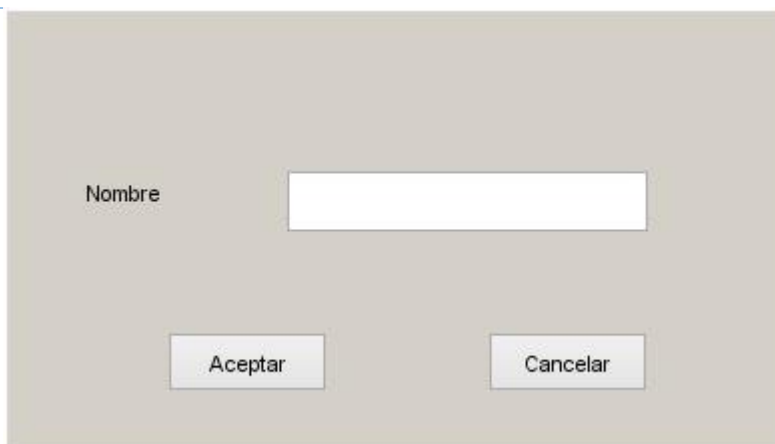


Figura 45: Interfaz gráfica de usuario

Tabla 30: Especificación de requisito Eliminar estructura de actividades.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y la estructura de actividades ha sido creada.
Flujo de eventos	
Flujo básico: Eliminar estructura de actividades.	
1.	Se da clic derecho sobre una estructura.
2.	Se selecciona la opción eliminar.
3	El sistema muestra un mensaje pidiendo la confirmación de eliminar.
4	Se acepta la confirmación.

5	El sistema elimina la estructura.	
6	Concluye el requisito	
Post-condiciones		
1	Se elimina la estructura de actividades.	
Flujo alternativo 4.a El usuario cancela la acción		
1	Concluye el requisito	
Post-condiciones		
1	No se elimina la estructura de actividades.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

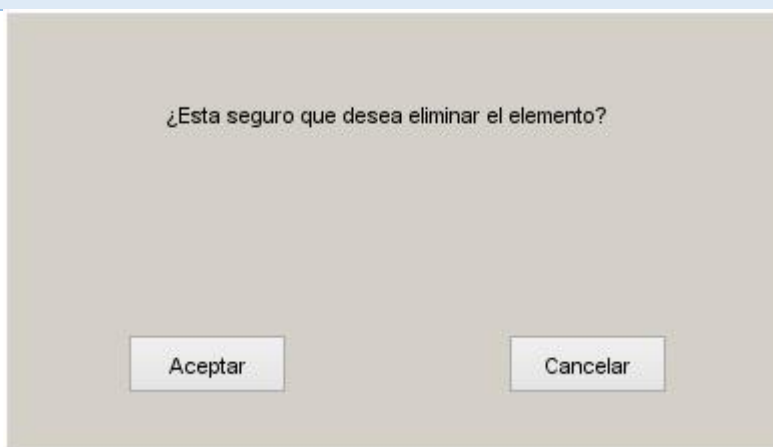


Figura 46: Interfaz gráfica de usuario

Tabla 31: Especificación de requisito Crear parte del rol.

Precondiciones	El usuario debe estar autenticado, el diseño de
-----------------------	--

	aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado un acto.	
Flujo de eventos		
Flujo básico: Crear parte del rol.		
1	Se selecciona la opción método.	
2	Se selecciona el elemento parte del rol	
3	Se da un clic en un acto creado.	
4	El sistema agrega la parte del rol al acto cliqueado.	
Post-condiciones		
1	Se Agrega la parte del rol al acto	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

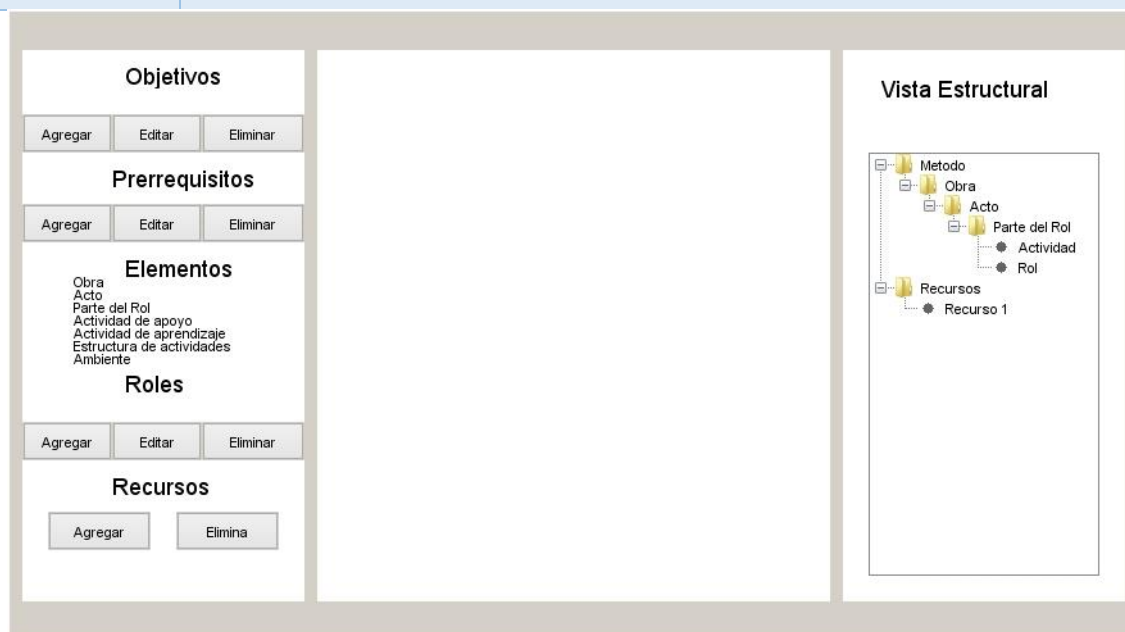


Figura 47: Interfaz gráfica de usuario

Tabla 32: Especificación de requisito Editar parte del rol.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y la parte del rol ha sido creada.
Flujo de eventos	
Flujo básico: Editar parte del rol.	
1	Se da clic derecho sobre la parte del rol a editar.
2	Se selecciona la opción editar información.
3	Se muestra un formulario con los siguientes campos con los datos de la parte del rol: <ul style="list-style-type: none"> • Nombre
4	Se editan los datos
5	Se selecciona la opción Aceptar.
6	El sistema valida los campos insertados
7	Si los datos son correctos el sistema edita la parte del rol
8	Concluye el requisito
Post-condiciones	
1	Se edita la parte del rol.
Flujos alternativos	
Flujo alternativo 6.a Datos incorrectos	
1	El sistema señala los datos incorrectos y permite corregirlos.
2	Se corrigen los datos.
3	Volver al paso 5 del flujo básico.
Post-condiciones	

1	N/A	
Flujo alternativo 5.a El usuario cancela la acción		
1	Concluye el requisito.	
Post-condiciones		
1	No se edita la parte del rol.	
Validaciones		
3.	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	



Figura 48: Interfaz gráfica de usuario

Tabla 33: Especificación de requisito Eliminar parte del rol.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y la parte del rol ha sido creada.
-----------------------	--

Flujo de eventos		
Flujo básico: Eliminar parte del rol.		
1.	Se da clic derecho sobre una parte del rol.	
2.	Se selecciona la opción eliminar.	
3	El sistema muestra un mensaje pidiendo la confirmación de eliminar.	
4	Se acepta la confirmación.	
5	El sistema elimina la parte del rol.	
6	Concluye el requisito	
Post-condiciones		
1	Se elimina la parte del rol.	
Flujo alternativo 4.a El usuario cancela la acción		
1	Concluye el requisito	
Post-condiciones		
1	No se elimina la parte del rol.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

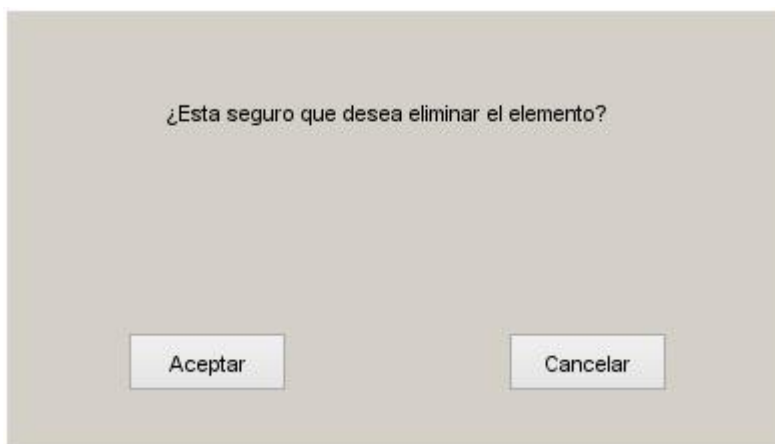


Figura 49: Interfaz gráfica de usuario

Tabla 34: Especificación de requisito Crear acto.

Precondiciones		El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una obra.
Flujo de eventos		
Flujo básico: Crear acto.		
1	Se selecciona la opción método.	
2	Se selecciona el elemento acto	
3	Se da un clic en una obra creada.	
4	El sistema agrega el acto a la obra cliqueada.	
Post-condiciones		
1	Se Agrega el acto a la obra.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	

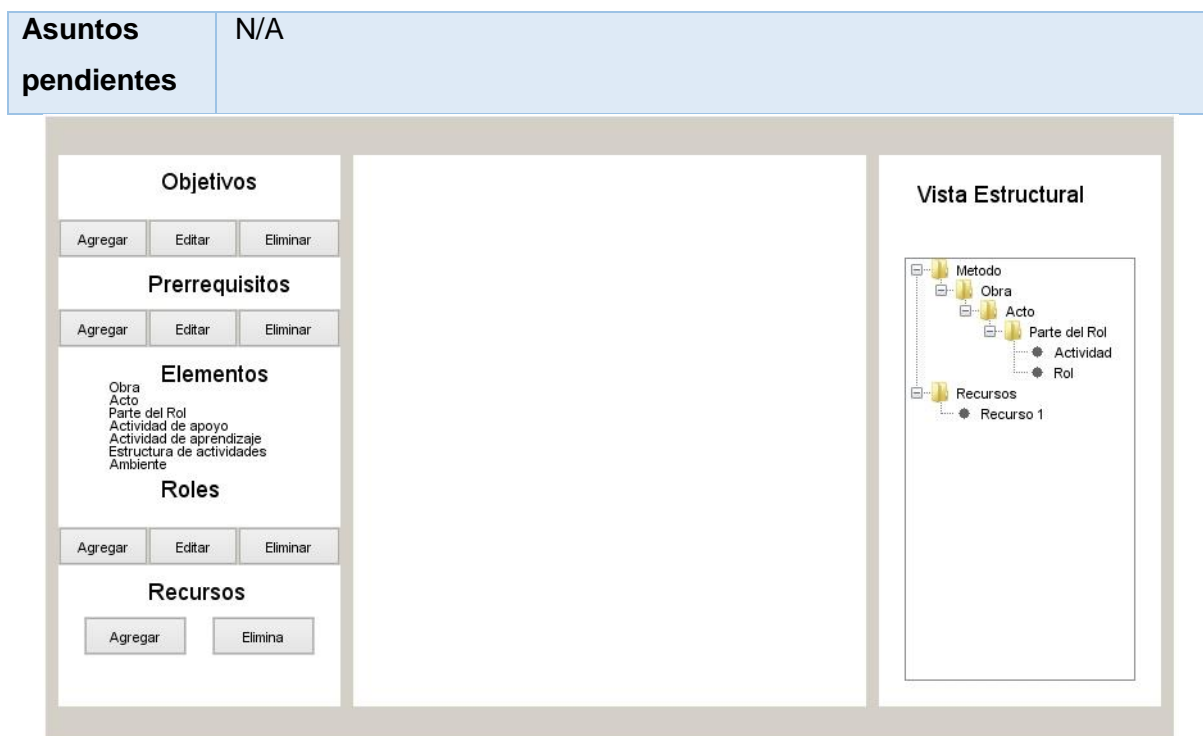


Figura 50: Interfaz gráfica de usuario

Tabla 35: Especificación de requisito Editar acto.

<p>Precondiciones</p>	<p>El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el acto ha sido creado.</p>
<p>Flujo de eventos</p>	
<p>Flujo básico: Editar acto.</p>	
<p>1</p>	<p>Se da clic derecho sobre el acto a editar.</p>
<p>2</p>	<p>Se selecciona la opción editar información.</p>
<p>3</p>	<p>Se muestra un formulario con los siguientes campos con los datos del acto:</p> <ul style="list-style-type: none"> • Descripción del acto. • Completamiento del acto.

	<ul style="list-style-type: none"> Retroalimentación del acto. 	
4	Se editan los datos	
5	Se selecciona la opción Aceptar.	
6	El sistema valida los campos insertados	
7	Si los datos son correctos el sistema edita el acto.	
8	Concluye el requisito	
Post-condiciones		
9	Se edita el acto.	
Flujos alternativos		
Flujo alternativo 6.a Datos incorrectos		
1	El sistema señala los datos incorrectos y permite corregirlos.	
2	Se corrigen los datos.	
3	Volver al paso 5 del flujo básico.	
Post-condiciones		
1	N/A	
Flujo alternativo 5.a El usuario cancela la acción		
1	Concluye el requisito.	
Post-condiciones		
1	No se edita el acto.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Figura 51: Interfaz gráfica de usuario

Tabla 36: Especificación de requisito Eliminar acto.

<p>Precondiciones</p>	<p>El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el acto ha sido creado.</p>
<p>Flujo de eventos</p>	

Flujo básico: Eliminar acto.		
1.	Se da clic derecho sobre un acto.	
2.	Se selecciona la opción eliminar.	
3	El sistema muestra un mensaje pidiendo la confirmación de eliminar.	
4	Se acepta la confirmación.	
5	El sistema elimina el acto.	
6	Concluye el requisito	
Post-condiciones		
1	Se elimina el acto.	
Flujo alternativo 4.a El usuario cancela la acción		
1	Concluye el requisito	
Post-condiciones		
1	No se elimina el acto.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

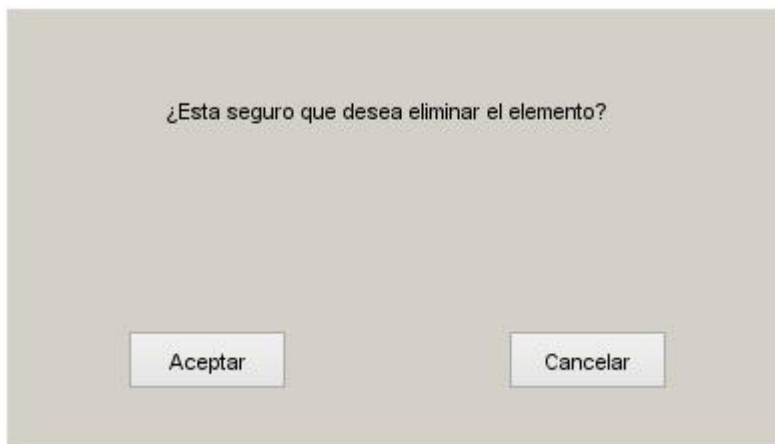


Figura 52: Interfaz gráfica de usuario

Tabla 37: Especificación de requisito Crear ambiente.

Precondiciones		El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Flujo de eventos		
Flujo básico: Crear obra.		
1	Se selecciona la opción método.	
2	Se selecciona el elemento obra	
3	Se da un clic sobre el panel de edición del método.	
4	El sistema agrega la obra.	
Post-condiciones		
1	Se Agrega la obra.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos	N/A	

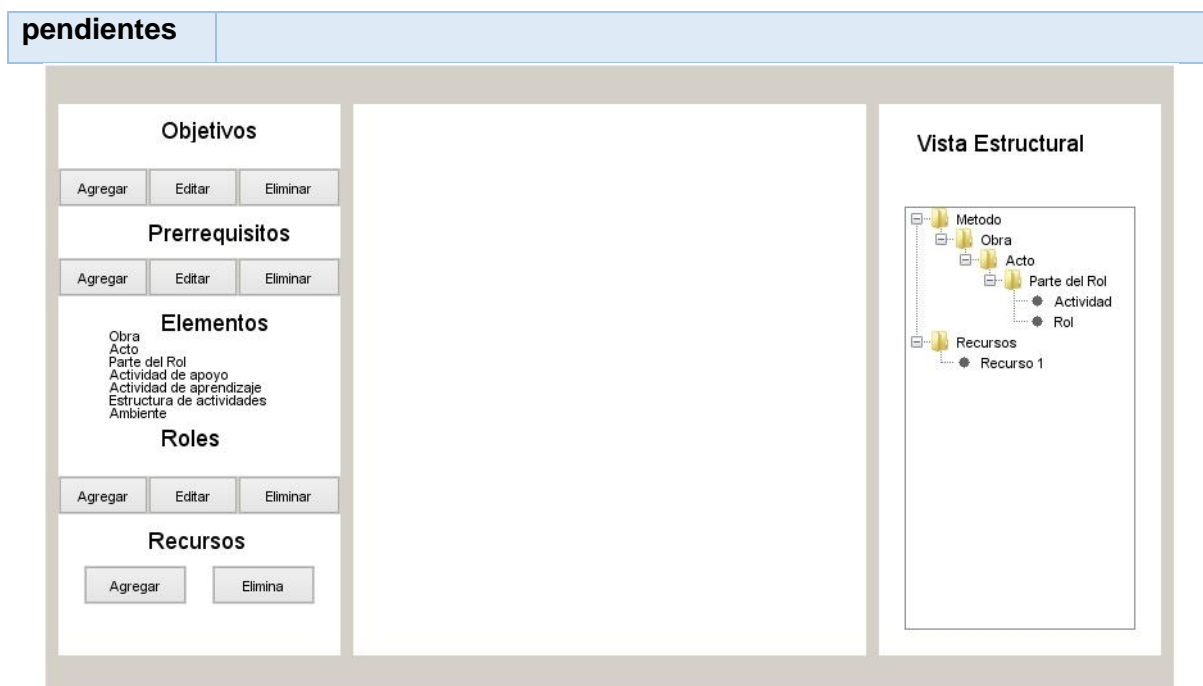


Figura 53: Interfaz gráfica de usuario

Tabla 38: Especificación de requisito Editar obra.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y la obra ha sido creada.
Flujo de eventos	
Flujo básico: Editar obra.	
1	Se da clic derecho sobre la obra a editar.
2	Se selecciona la opción editar información.
3	Se muestra un formulario con los siguientes campos con los datos de la obra: <ul style="list-style-type: none"> • Descripción del acto. • Completamiento del acto. • Retroalimentación del acto.

4	Se editan los datos	
5	Se selecciona la opción Aceptar.	
6	El sistema valida los campos insertados	
7	Si los datos son correctos el sistema edita la obra.	
8	Concluye el requisito	
Post-condiciones		
9	Se edita la obra.	
Flujos alternativos		
Flujo alternativo 6.a Datos incorrectos		
1	El sistema señala los datos incorrectos y permite corregirlos.	
2	Se corrigen los datos.	
3	Volver al paso 5 del flujo básico.	
Post-condiciones		
1	N/A	
Flujo alternativo 5.a El usuario cancela la acción		
1	Concluye el requisito.	
Post-condiciones		
1	No se edita la obra.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Figura 54: Interfaz gráfica de usuario

Tabla 39: Especificación de requisito Eliminar obra.

<p>Precondiciones</p>	<p>El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y la obra ha sido creada.</p>
<p>Flujo de eventos</p>	

Flujo básico: Eliminar obra.		
1.	Se da clic derecho sobre una obra.	
2.	Se selecciona la opción eliminar.	
3	El sistema muestra un mensaje pidiendo la confirmación de eliminar.	
4	Se acepta la confirmación.	
5	El sistema elimina la obra.	
6	Concluye el requisito	
Post-condiciones		
1	Se elimina la obra.	
Flujo alternativo 4.a El usuario cancela la acción		
1	Concluye el requisito	
Post-condiciones		
1	No se elimina la obra.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

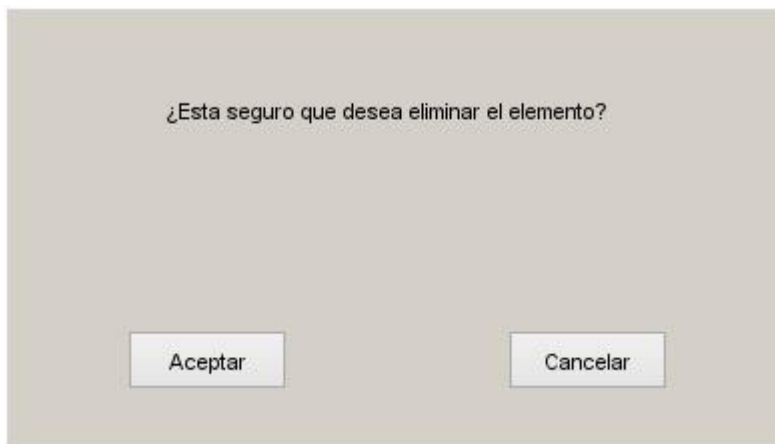


Figura 55: Interfaz gráfica de usuario

Tabla 40: Especificación de requisito Completar método.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Flujo de eventos	
Flujo básico: Completar método.	
1	Se da clic derecho sobre el método.
2	Se selecciona la opción editar información.
3	Se muestra un formulario con los siguientes campos con los datos de la obra: <ul style="list-style-type: none"> • Completamiento del método. • Retroalimentación del método.
4	Se editan los datos
5	Se selecciona la opción Aceptar.
6	El sistema valida los campos insertados
7	Si los datos son correctos el sistema edita el método.
8	Concluye el requisito

Post-condiciones		
9	Se edita el método.	
Flujos alternativos		
Flujo alternativo 6.a Datos incorrectos		
1	El sistema señala los datos incorrectos y permite corregirlos.	
2	Se corrigen los datos.	
3	Volver al paso 5 del flujo básico.	
Post-condiciones		
1	N/A	
Flujo alternativo 5.a El usuario cancela la acción		
1	Concluye el requisito.	
Post-condiciones		
1	No se edita el método.	
Validaciones		
1	Se validan los datos.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Figura 56: Interfaz gráfica de usuario

Tabla 41: Especificación de requisito Agregar recurso.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado y se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Flujo de eventos	
Flujo básico: Agregar recurso.	
1	Se selecciona la opción Recurso.
2	Se selecciona la opción agregar nuevo recurso
3	Se muestra la ventana de adición de recursos.

4	Se selecciona la opción examinar.	
5	El sistema muestra una ventana de explorador del sistema de archivos.	
6	Se selecciona el archivo a agregar como recurso	
7	Se agrega el recurso.	
8	Concluye el requisito.	
Post-condiciones		
9	Se Agrega un recurso	
Flujos alternativos		
Flujo alternativo 6.a Imposible subir el recurso		
1	El sistema señala que es imposible subir el recurso.	
2	Volver al paso 4 del flujo básico.	
Post-condiciones		
1	N/A	
Flujo alternativo 4.a El usuario cancela la acción		
1	Concluye el requisito.	
Post-condiciones		
1	No agrega el recurso	
Validaciones		
1	Se valida el tamaño del archivo y que este no este agregado ya.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> Visibles en la interfaz: <ul style="list-style-type: none"> Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

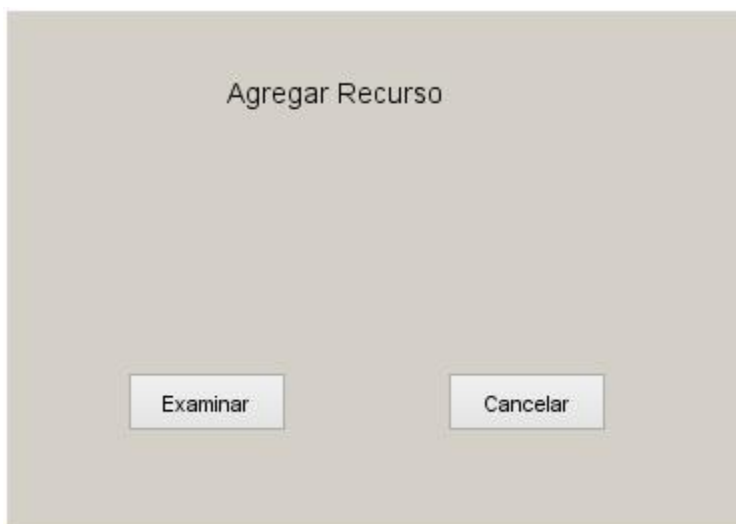


Figura 57: Interfaz gráfica de usuario

Tabla 42: Especificación de requisito Eliminar recurso.

Precondiciones	El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje.	
Flujo de eventos		
Flujo básico: Eliminar recurso.		
1	Se selecciona la opción recursos.	
2	Se selecciona un recurso.	
3	Se selecciona el botón eliminar recurso.	
5	El sistema elimina el recurso.	
6	Concluye el requisito	
Post-condiciones		
1	Se elimina el recurso.	
Conceptos	Usuario	Atributos del concepto que se utilizan en el requisito: <ul style="list-style-type: none"> • Visibles en la interfaz:

		Nombre de usuario
Requisitos especiales	N/A	
Asuntos pendientes	N/A	

Anexo Casos de prueba de aceptación:

Tabla 43: CPA Editar diseño de aprendizaje.

Caso de prueba de aceptación
Especificación de requisito: Editar diseño de aprendizaje
Nombre : Editar diseño de aprendizaje
Descripción: Edita un diseño de aprendizaje.
Condiciones de ejecución: El usuario debe estar autenticado y el diseño de aprendizaje ha sido creado.
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje <input type="checkbox"/> Seleccionar la opción editar. <input type="checkbox"/> Se selecciona la opción editar diseño de aprendizaje. <input type="checkbox"/> Se llenan los campos correspondientes. <input type="checkbox"/> Se da presiona en el botón guardar.
Resultado esperado: El edita la información de un diseño de aprendizaje.
Evaluación de prueba: Prueba satisfactoria.

Tabla 44: CPA Crear ambiente.

Caso de prueba de aceptación

Especificación de requisito: Crear ambiente.
Nombre : Crear ambiente del diseño de aprendizaje
Descripción: Adiciona un ambiente a una parte del rol.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una parte del rol.
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Seleccionar la opción método. <input type="checkbox"/> Seleccionar la opción ambiente. <input type="checkbox"/> Dar clic sobre una parte del rol.
Resultado esperado: El sistema adiciona un ambiente a la parte del rol seleccionada.
Evaluación de prueba: Prueba satisfactoria.

Tabla 45: CPA Editar ambiente.

Caso de prueba de aceptación
Especificación de requisito: Editar ambiente.
Nombre: Editar ambiente del diseño de aprendizaje.
Descripción: Edita un ambiente.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado un ambiente.
Pasos de ejecución:

<ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Se da clic derecho sobre un ambiente. <input type="checkbox"/> Se selecciona la opción editar información. <input type="checkbox"/> Se llenan los campos correspondientes. <input type="checkbox"/> Se selecciona la opción Aceptar
Resultado esperado: El edita la información de un ambiente.
Evaluación de prueba: Prueba satisfactoria.

Tabla 46: CPA Crear roles.

Caso de prueba de aceptación
Especificación de requisito: Crear roles.
Nombre: Crear roles.
Descripción: Adiciona un rol.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado y se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Seleccionar la opción roles. <input type="checkbox"/> Seleccionar el botón agregar nuevo rol. <input type="checkbox"/> Se llenan los campos necesarios <input type="checkbox"/> Seleccionar la opción Agregar.
Resultado esperado: El sistema adiciona un rol.
Evaluación de prueba: Prueba satisfactoria.

Tabla 47: CPA Editar roles.

Caso de prueba de aceptación	
Código:	Especificación de requisito: Editar roles
Nombre: Editar roles.	
Descripción: Edita un rol.	
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el rol ha sido creado.	
Pasos de ejecución:	
<ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Se selecciona la opción roles. <input type="checkbox"/> Seleccionar un rol. <input type="checkbox"/> Seleccionar el botón Editar rol. <input type="checkbox"/> Se selecciona la opción Aceptar 	
Resultado esperado: El edita la información de un rol.	
Evaluación de prueba: Prueba satisfactoria.	

Tabla 48: CPA Crear prerrequisito.

Caso de prueba de aceptación	
Código:	
Especificación de requisito: Crear prerrequisitos.	

Nombre: Crear prerrequisitos.
Descripción: Adiciona un prerrequisitos.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado y se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Pasos de ejecución: <ul style="list-style-type: none"><input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido.<input type="checkbox"/> Seleccionar la opción prerrequisitos.<input type="checkbox"/> Seleccionar el botón agregar nuevo prerrequisito.<input type="checkbox"/> Se llenan los campos necesarios.<input type="checkbox"/> Seleccionar la opción Aceptar.
Resultado esperado: El sistema adiciona un prerrequisito.
Evaluación de prueba: Prueba satisfactoria.

Tabla 49: CPA Editar prerrequisito.

Caso de prueba de aceptación
Especificación de requisito: Editar prerrequisito.
Nombre: Editar prerrequisito.
Descripción: Edita un prerrequisito.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el prerrequisito ha sido creado.
Pasos de ejecución: <ul style="list-style-type: none"><input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido.<input type="checkbox"/> Se selecciona la opción prerrequisito.

<ul style="list-style-type: none"> <input type="checkbox"/> Seleccionar un prerrequisito. <input type="checkbox"/> Seleccionar el botón Editar prerrequisito. <input type="checkbox"/> Se llenan los campos correspondientes. <input type="checkbox"/> Se selecciona la opción Aceptar
Resultado esperado: El edita la información de un rol.
Evaluación de prueba: Prueba satisfactoria.

Tabla 50: CPA Crear objetivo general.

Caso de prueba de aceptación
Especificación de requisito: Crear objetivo general.
Nombre: Crear objetivo general.
Descripción: Adiciona un objetivo general.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado y se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Seleccionar la opción objetivos generales. <input type="checkbox"/> Seleccionar el botón agregar nuevo objetivo general. <input type="checkbox"/> Se llenan los campos correspondientes. <input type="checkbox"/> Seleccionar la opción Aceptar.
Resultado esperado: El sistema adiciona un prerrequisito.
Evaluación de prueba: Prueba satisfactoria.

Tabla 51: CPA Editar objetivos generales.

Caso de prueba de aceptación
Especificación de requisito: Editar objetivos generales.
Nombre: Editar objetivos generales.
Descripción: Edita un objetivo general.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el objetivo ha sido creado.
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Se selecciona la opción objetivos generales. <input type="checkbox"/> Seleccionar un objetivo general. <input type="checkbox"/> Seleccionar el botón Editar objetivo general. <input type="checkbox"/> Se llenan los campos correspondientes. <input type="checkbox"/> Se selecciona la opción Aceptar
Resultado esperado: El edita la información de un rol.
Evaluación de prueba: Prueba satisfactoria.

Tabla 52: CPA Crear actividad de aprendizaje.

Caso de prueba de aceptación
Especificación de requisito: Crear actividad de aprendizaje
Nombre : Crear actividad de aprendizaje
Descripción: Adiciona una actividad a una parte del rol.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una parte del rol.

<p>Pasos de ejecución:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear actividad de aprendizaje/creación del contenido o editar actividad de aprendizaje/edición del contenido. <input type="checkbox"/> Seleccionar la opción método. <input type="checkbox"/> Seleccionar la opción actividad de aprendizaje. <input type="checkbox"/> Dar clic sobre una parte del rol.
<p>Resultado esperado: El sistema adiciona una actividad a la parte del rol seleccionada.</p>
<p>Evaluación de prueba: Prueba satisfactoria.</p>

Tabla 53: CPA Editar actividad de aprendizaje.

Caso de prueba de aceptación
<p>Especificación de requisito: Editar actividad de aprendizaje</p>
<p>Nombre : Editar actividad de aprendizaje</p>
<p>Descripción: Edita una actividad de aprendizaje.</p>
<p>Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una actividad.</p>
<p>Pasos de ejecución:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Se da clic derecho sobre una actividad. <input type="checkbox"/> Se selecciona la opción editar información. <input type="checkbox"/> Se llenan los campos correspondientes. <input type="checkbox"/> Se selecciona la opción Aceptar
<p>Resultado esperado: El edita la información de una actividad.</p>
<p>Evaluación de prueba: Prueba satisfactoria.</p>

Tabla 54: CPA Crear actividad de apoyo.

Caso de prueba de aceptación
Especificación de requisito: Crear actividad de apoyo.
Nombre: Crear actividad de apoyo.
Descripción: Adiciona una actividad de apoyo a una parte del rol.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una parte del rol.
Pasos de ejecución: <ul style="list-style-type: none"><input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear actividad de apoyo de aprendizaje/creación del contenido o editar actividad de apoyo de aprendizaje/edición del contenido.<input type="checkbox"/> Seleccionar la opción método.<input type="checkbox"/> Seleccionar la opción actividad de apoyo.<input type="checkbox"/> Dar clic sobre una parte del rol.
Resultado esperado: El sistema adiciona una actividad de apoyo a la parte del rol seleccionada.
Evaluación de prueba: Prueba satisfactoria.

Tabla 55: CPA Editar actividad de apoyo.

Caso de prueba de aceptación
Especificación de requisito: Editar actividad de apoyo.
Nombre: Editar actividad de apoyo.
Descripción: Edita una actividad de apoyo.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una actividad.
Pasos de ejecución: <ul style="list-style-type: none"><input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido.<input type="checkbox"/> Se da clic derecho sobre una actividad de apoyo.<input type="checkbox"/> Se selecciona la opción editar información.<input type="checkbox"/> Se llenan los campos correspondientes.<input type="checkbox"/> Se selecciona la opción Aceptar
Resultado esperado: El edita la información de una actividad de apoyo.
Evaluación de prueba: Prueba satisfactoria.

Tabla 56: CPA Crear parte del rol.

Caso de prueba de aceptación
Especificación de requisito: Crear parte del rol

Nombre : Crear parte del rol
Descripción: Adiciona parte del rol.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado un acto.
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Seleccionar la opción método. <input type="checkbox"/> Seleccionar la opción parte del rol. <input type="checkbox"/> Dar clic en un acto.
Resultado esperado: El sistema adiciona una parte del rol.
Evaluación de prueba: Prueba satisfactoria.

Tabla 57: CPA Editar parte del rol.

Caso de prueba de aceptación
Especificación de requisito: Editar parte del rol.
Nombre: Editar parte del rol.
Descripción: Edita parte del rol.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y la parte del rol ha sido creada
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Se da clic derecho sobre una parte del rol. <input type="checkbox"/> Se selecciona la opción renombrar elemento.

<ul style="list-style-type: none"><input type="checkbox"/> Se llenan los campos correspondientes.<input type="checkbox"/> Se selecciona la opción Aceptar
Resultado esperado: El edita la información de una parte de rol.
Evaluación de prueba: Prueba satisfactoria.

Tabla 58: CPA Crear acto.

Caso de prueba de aceptación
Especificación de requisito: Crear acto.
Nombre: Crear acto.
Descripción: Adiciona un acto a una obra.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y se ha creado una obra.
Pasos de ejecución: <ul style="list-style-type: none"><input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido.<input type="checkbox"/> Seleccionar la opción método.<input type="checkbox"/> Seleccionar la opción acto de aprendizaje.<input type="checkbox"/> Dar clic sobre una obra.
Resultado esperado: El sistema adiciona un acto a la obra seleccionada.
Evaluación de prueba: Prueba satisfactoria.

Tabla 59: CPA Editar acto.

Caso de prueba de aceptación
Especificación de requisito: Editar acto.

Nombre: Editar acto.
Descripción: Edita un acto.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y el acto ha sido creado.
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Se da clic derecho sobre un acto. <input type="checkbox"/> Se selecciona la opción editar información. <input type="checkbox"/> Se llenan los campos correspondientes. <input type="checkbox"/> Se selecciona la opción Aceptar
Resultado esperado: El sistema edita la información de un acto.
Evaluación de prueba: Prueba satisfactoria.

Tabla 60: CPA Crear obra.

Caso de prueba de aceptación
Especificación de requisito: Crear obra.
Nombre: Crear obra.
Descripción: Adiciona una obra.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Seleccionar la opción método.

<ul style="list-style-type: none"> <input type="checkbox"/> Seleccionar la opción obra de aprendizaje. <input type="checkbox"/> Dar clic sobre el panel de edición del método.
Resultado esperado: El sistema adiciona una obra al método.
Evaluación de prueba: Prueba satisfactoria.

Tabla 61: CPA Editar obra.

Caso de prueba de aceptación
Especificación de requisito: Editar obra
Nombre: Editar obra.
Descripción: Edita obra.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado, se encuentra en la vista de creación del contenido del diseño de aprendizaje y la obra ha sido creada.
Pasos de ejecución: <ul style="list-style-type: none"> <input type="checkbox"/> Ir a la lista de diseños de aprendizaje/crear diseño de aprendizaje/creación del contenido o editar diseño de aprendizaje/edición del contenido. <input type="checkbox"/> Se da clic derecho sobre una obra. <input type="checkbox"/> Se selecciona la opción editar información. <input type="checkbox"/> Se llenan los campos correspondientes. <input type="checkbox"/> Se selecciona la opción Aceptar
Resultado esperado: El sistema edita la información de una obra.
Evaluación de prueba: Prueba satisfactoria.

Tabla 62: CPA Agregar recurso.

Caso de prueba de aceptación
Especificación de requisito: Agregar recurso
Nombre : Agregar recurso
Descripción: Adiciona un recurso al diseño de aprendizaje.
Condiciones de ejecución: El usuario debe estar autenticado, el diseño de aprendizaje ha sido creado y se encuentra en la vista de creación del contenido del diseño de aprendizaje.
Pasos de ejecución: <ul style="list-style-type: none"><input type="checkbox"/> Seleccionar la opción recurso.<input type="checkbox"/> Seleccionar la opción agregar nuevo recurso.<input type="checkbox"/> Seleccionar la opción examinar.<input type="checkbox"/> Se selecciona el archivo a agregar como recurso.
Resultado esperado: El sistema adiciona un recurso al diseño de aprendizaje.
Evaluación de prueba: Prueba satisfactoria.