



Universidad de las Ciencias Informáticas
“Facultad 4”

**Implementación de IMS QTI para la realización de
cuestionarios en CRODA.**

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

Autor:

Roberto Hamlet Leyva Ramos.

Tutores:

Jorge Iturria Pozo.

Oswaldo Ernesto Stable Vilches.

Aliuska Sánchez Ibarria.

Ciudad de la Habana, junio 2011

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Roberto Hamlet Leyva Ramos

Firma del Autor

Jorge Iturria Pozo

Firma del Tutor

Oswaldo Ernesto Stable Vilches

Firma del Tutor

Aliuska Sánchez Ibarria

Firma del Tutor

Agradecimientos

Les agradezco a todas las personas que hicieron posible la realización de este trabajo.

A mis padres Carmen Y Roberto que siempre han estado a mi lado dándome su apoyo y amor.

A mis tutores Iturria y Osvaldo por brindarme su ayuda incondicional en todo momento.

A todos mis familiares queridos, abuelos, tíos y primos que de una forma u otra me han apoyado siempre.

A mis amigos de la universidad, Mylen, Leonel, Wisel, Reinier.

A mis otros amigos, Luis Manuel, Jose Carlos, Yenier, Maiquel.

Un agradecimiento especial a mi querida abuela María del Pilar, que siempre soñó con verme hecho un profesional, llegue a ti donde quieras que estés mi mayor agradecimiento y mejor regalo, tu sueño realizado.

Dedicatoria

*A mis padres por entregarme todo su amor y apoyo que me guiaron a alcanzar mis
sueños.*

RESUMEN

La forma de educar durante el transcurso del tiempo ha sido objeto de cambios, donde la tendencia actual es hacia una educación virtualizada. Aquí es donde se ponen de manifiesto los diferentes tipos de herramientas de autor y plataformas que permiten hacer realidad este tipo de enseñanza. En la actualidad gran parte de las plataformas de aprendizaje incluyen con menor o mayor funcionalidad una herramienta para la creación, gestión y realización de exámenes en línea. Estos son utilizados ampliamente como herramienta para evaluar si un alumno ha asimilado los conceptos que le han sido presentados y como herramienta de autoevaluación. En este sentido, el uso de herramientas informáticas ha supuesto un avance, ya que un gran número de tipos de preguntas pueden ser corregidas de manera automática mediante el uso de este tipo de herramientas.

El presente trabajo tiene como objetivo implementar un módulo de la herramienta de autor CRODA, que haciendo uso de la especificación IMS QTI permita crear preguntas que puedan ser reutilizables por cualquier plataforma de aprendizaje o herramienta de autoría con soporte a dicha especificación.

Como resultado del presente trabajo se logró la creación de preguntas reutilizables e interoperables, generándose los artefactos correspondientes al flujo de implementación y pruebas. El sistema fue validado con la técnica de caja negra.

Palabras claves: Herramienta de autor, LMS, objeto de aprendizaje, e-learning, estándar, especificación, IMS-QTI, assessment, item, cuestionario, pregunta, interacción.

ÍNDICE DE CONTENIDO

| | |
|---|----|
| INTRODUCCIÓN | 1 |
| CAPÍTULO 1: Fundamentación teórica | 4 |
| 1.1 Introducción | 4 |
| 1.2 La enseñanza virtual (e-learning)..... | 4 |
| 1.3 Objetos de aprendizaje | 5 |
| 1.4 Sistema de gestión del aprendizaje (LMS) | 5 |
| 1.5 Sistema de gestión de contenidos educativos (LCMS) | 6 |
| 1.6 Herramientas de autor | 6 |
| 1.6.1 Respondus | 7 |
| 1.6.2 eXeLearning | 7 |
| 1.6.3 Reload | 8 |
| 1.6.4 CRODA..... | 8 |
| 1.7 Estándares y especificaciones en e-learning | 8 |
| 1.7.1 Ventajas aportadas por los estándares en e-learning | 9 |
| 1.8 IMS question & test interoperability specification | 10 |
| 1.8.1 Versiones de la especificación | 11 |
| 1.8.2 Perfiles de la especificación | 12 |
| 1.8.3 Conceptos básicos de IMS QTI V 2.0 | 13 |
| 1.8.3.1 Las preguntas en IMS QTI..... | 13 |
| 1.8.3.2 Las interacciones en IMS QTI..... | 14 |
| 1.8.4 Herramientas relacionadas con la especificación IMS QTI..... | 21 |
| 1.9 Herramientas y tecnologías a utilizar | 23 |
| 1.9.1 Metodología de desarrollo de software | 23 |
| 1.9.1.1 Extreme programming (XP) | 23 |
| 1.9.1.2 Proceso unificado de desarrollo (RUP) | 23 |
| 1.9.2 Tecnologías a utilizar en la Web | 24 |

| | | |
|----------------------------------|--|----|
| 1.9.3 | Procesador de hipertexto (PHP) | 26 |
| 1.9.4 | Framework..... | 27 |
| 1.9.4.1 | Symfony..... | 27 |
| 1.9.4.2 | ExtJS | 28 |
| 1.9.5 | Sistema gestor de base de datos (SGBD) | 29 |
| 1.9.5.1 | Oracle..... | 29 |
| 1.9.5.2 | MySQL | 29 |
| 1.9.5.3 | PostgreSQL..... | 30 |
| 1.9.6 | Servidor web..... | 30 |
| 1.9.6.1 | Zope | 30 |
| 1.9.6.2 | Internet information services (IIS) | 31 |
| 1.9.6.3 | Apache | 31 |
| 1.9.7 | Entorno de desarrollo integrado (IDE)..... | 31 |
| 1.9.7.1 | Eclipse..... | 32 |
| 1.9.7.2 | NetBeans..... | 32 |
| 1.10 | Conclusiones | 33 |
| CAPÍTULO 2: Implementación | | 34 |
| 2.1 | Introducción | 34 |
| 2.2 | Requisitos funcionales | 34 |
| 2.3 | Modelo de implementación | 36 |
| 2.3.1 | Diagrama de paquetes..... | 37 |
| 2.3.2 | Diagrama de componentes | 38 |
| 2.3.2.1 | Funcionalidades principales..... | 41 |
| 2.4 | Modelo de despliegue..... | 45 |
| 2.5 | Modelo de datos | 45 |
| 2.5.1 | Modelo de base de datos relacional..... | 46 |

| | | |
|---|---|----|
| 2.6 | Estándar de codificación | 47 |
| 2.7 | Conclusión | 49 |
| CAPÍTULO 3: Validación de la solución propuesta. | | 50 |
| 3.1 | Introducción | 50 |
| 3.2 | Técnicas de pruebas..... | 50 |
| 3.3 | Casos de prueba | 51 |
| 3.3.1 | Descripción de los casos de prueba. | 51 |
| 3.4 | Clasificación de las no conformidades | 58 |
| 3.5 | Resultados de las pruebas de caja negra. | 60 |
| 3.6 | Conclusiones | 60 |
| CONCLUSIONES | | 61 |
| RECOMENDACIONES | | 62 |
| REFERENCIAS BIBLIOGRÁFICAS | | 63 |
| Anexo 1. XML de los tipos de preguntas contemplados. | | 66 |
| GLOSARIO DE TÉRMINOS..... | | 69 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1.1. Ejemplo de choiceInteraction (selección simple) | 15 |
| Figura 1.2. Ejemplo de choiceInteraction (selección múltiple) | 15 |
| Figura 1.3. Ejemplo de ordeInteraction. | 16 |
| Figura 1.4. Ejemplo de associateInteraction..... | 17 |
| Figura 1.5. Ejemplo de matchInteraction..... | 17 |
| Figura 1.6. Ejemplo de gapMatchInteraction. | 18 |
| Figura 1.7. Ejemplo de inlineChoiceInteraction. | 19 |
| Figura 1.8. Ejemplo de textEntryInteraction..... | 19 |
| Figura 1.9. Ejemplo de extendedTextInteraction. | 20 |
| Figura 1.10. Ejemplo de hottextInteraction. | 20 |
| Figura 2.1. Diagrama de paquetes general del sistema. | 37 |
| Figura 2.2. Diagrama de paquetes de EXT JS. | 38 |
| Figura 2.3. Diagrama de componentes general del sistema..... | 39 |
| Figura 2.4. Diagrama de componentes general del módulo. | 40 |
| Figura 2.5. Diagrama de componentes mostrar ejercicios..... | 41 |
| Figura 2.6. Diagrama de componentes crear nueva pregunta..... | 42 |
| Figura 2.7. Diagrama de componentes gestionar pregunta. | 43 |
| Figura 2.8. Diagrama de componentes exportar ejercicio. | 44 |
| Figura 2.9. Diagrama de componentes exportar pregunta. | 44 |
| Figura 2.10. Diagrama de despliegue. | 45 |
| Figura 2.11. Diagrama entidad-relación del módulo ejercicios. | 47 |
| Figura 2.12. Nomenclatura de los identificadores..... | 48 |
| Figura 2.13. Normas de los comentarios en las clases. | 48 |
| Figura 2.14. Normas de los comentarios en las funciones. | 49 |

| | |
|--|----|
| Figura 2.15. Identación del código. | 49 |
| Figura A1. XML preguntas de verdadero o falso. | 66 |
| Figura A2. XML preguntas de selección simple..... | 67 |
| Figura A3. XML preguntas de respuestas múltiples. | 67 |
| Figura A4. XML preguntas de espacios en blanco. | 68 |
| Figura A5. XML preguntas de enlazar columna..... | 68 |

ÍNDICE DE TABLAS

| | |
|--|----|
| Tabla 1. Escenarios para el caso de uso “Gestionar ejercicio” | 52 |
| Tabla 2. Caso de prueba “Nuevo ejercicio” | 53 |
| Tabla 3. Caso de prueba “Previsualizar ejercicio” | 53 |
| Tabla 4. Caso de prueba “Exportar ejercicio” | 53 |
| Tabla 5. Caso de prueba “Eliminar ejercicio” | 54 |
| Tabla 6. Caso de prueba “Editar ejercicio” | 54 |
| Tabla 7. Partición de equivalencia. | 54 |
| Tabla 8. Escenarios para el caso de uso “Gestionar preguntas de verdadero o falso” | 55 |
| Tabla 9. Caso de prueba “Nueva pregunta de verdadero o falso” | 56 |
| Tabla 10. Caso de prueba “Editar pregunta de verdadero o falso” | 57 |
| Tabla 11. Caso de prueba “Eliminar pregunta” | 57 |
| Tabla 12. Caso de prueba “Exportar pregunta” | 58 |
| Tabla 13. Caso de prueba “Previsualizar pregunta” | 58 |
| Tabla 14. Resultado de las pruebas funcionales. | 60 |

INTRODUCCIÓN

Los ordenadores se han empleado en tareas educativas prácticamente desde su aparición a mediados del siglo XX, a partir de la generalización de internet como medio de comunicación es cuando se ha producido una revolución que está teniendo un impacto real en la educación. La forma de educar está cambiando ya que cada vez más se utilizan los medios informáticos como un complemento a las clases presenciales. (Baltasar Fernández Manjón, 2006)

En la actualidad es imposible pensar en el acceso a la información y en la gestión del conocimiento sin el uso y la aplicación de las nuevas tecnologías de la información y la comunicación (TIC). El desarrollo de Internet, y particularmente de herramientas cada vez más eficientes para la comunicación y la interacción humana, posibilitan el aprendizaje activo y colaborador en red, con la creación de espacios virtuales que se integran a los procesos docentes de carácter presencial. Es una nueva dimensión formativa que está cambiando no solo los tradicionales medios de enseñanza en herramientas para el aprendizaje, sino lo más importante: la comunicación durante el proceso de enseñanza-aprendizaje y su propia dinámica. (E-learning en Cuba: un ejemplo y un reto para la epidemiología cu, 2005)

En Cuba también se han hecho sentir estos cambios, donde diversas instituciones educativas presenciales han comenzado a penetrar en el desarrollo de modalidades no presenciales o semipresenciales basados en las redes de comunicación o en otras tecnologías, tales como las videoconferencias, etc. Actualmente el modelo pedagógico que se utiliza en las sedes universitarias municipales utiliza las modalidades a distancia y semipresencial.

La Universidad de las Ciencias Informáticas cuenta con un Centro de Tecnologías para la Formación (FORTES) que desarrolla tecnologías que permiten ofrecer servicios y productos para la implementación de soluciones de formación. Este centro FORTES cuenta con varios proyectos, de los cuales uno de ellos se encarga del desarrollo de la herramienta de autor CRODA, que permite la creación de objetos de aprendizaje (OA) reutilizables, accesibles, duraderos e interoperables, de forma flexible, empleando los estándares SCORM y LOM. La herramienta en la versión 1.0 cuenta con un módulo para la creación de cuestionarios, que no se rige por ninguna especificación o estándar que permita la interoperabilidad y reusabilidad de los mismos, de forma que estos puedan ser importados en cualquier LMS o herramienta de autoría con soporte a dicho estándar. Estos cuestionarios requieren de cierto esfuerzo para su

elaboración, el cual puede llegar a perderse si fuera necesario cambiar de plataforma de aprendizaje, por tanto sería deseable tener la posibilidad de compartir este esfuerzo permitiendo el intercambio de cuestionarios.

Por todo lo anteriormente planteado se presenta como **problema a resolver**: ¿Cómo desarrollar cuestionarios en CRODA que logren una interacción con las herramientas educativas empleando como estándar IMS QTI? Para brindarle una solución a este problema se define como **objeto de estudio** el desarrollo y funcionamiento de cuestionarios que hagan uso del estándar IMS QTI para lograr la interacción y reutilización de ellos en las herramientas de autor y LMS.

El **campo de acción** de esta investigación se enmarca en la implementación de IMS QTI para la creación de cuestionarios en CRODA.

En correspondencia a lo antes expuesto se propone como **objetivo general**, implementar las funcionalidades que permitan la realización de cuestionarios con la utilización de IMS QTI, en la herramienta de autor CRODA. Para cumplir con este objetivo se plantean los siguientes **objetivos específicos**:

- Estudiar herramientas, tecnologías, metodologías y la especificación IMS Question & Test Interoperability (IMS QTI) para desarrollar el módulo de ejercicios.
- Implementar los componentes necesarios para la realización de las preguntas (verdadero/falso, selección múltiple, respuestas múltiples, rellenar espacios en blanco y respuesta corta) con la utilización de IMS QTI en CRODA.
- Comprobar la correcta creación de cuestionarios en CRODA con la utilización de IMS QTI.

En función de la investigación se plantea como **idea a defender** con la implementación de funcionalidades en la herramienta de autor CRODA, que permitan la creación de cuestionarios basados en el estándar IMS QTI, se logrará el desarrollo de componentes que posibiliten la generación de cuestionarios estandarizados.

Los métodos científicos que se utilizan para llevar a cabo el desarrollo de la investigación son:

El **Analítico-Sintético**: se aplica en el presente trabajo mientras se realiza una investigación que incluye lectura y análisis de abundante documentación relacionada con las aplicaciones y técnicas existentes que permiten la creación de preguntas.

El **Histórico-lógico**: su empleo permite estudiar el estado y la evolución en los últimos años de las tendencias y técnicas existentes que posibiliten la creación de preguntas.

Modelación: su aplicación permite el esbozo de los diferentes diagramas y modelos generados durante el proceso de desarrollo de la aplicación.

El presente trabajo de diploma presenta la siguiente Estructura Capitular:

Capítulo 1.- Fundamentación Teórica: este capítulo contiene un estudio del estado del arte de las herramientas, tecnologías y metodologías utilizadas en la actualidad para brindar soluciones a este tipo de problemas. Este análisis posibilita la recopilación de gran cantidad de información que se utiliza para definir las librerías o framework y metodología usadas para la implementación del módulo.

Capítulo 2.- Implementación: en el presente capítulo se lleva a cabo la descripción de la implementación del módulo generándose los artefactos correspondientes. Se explican las principales funcionalidades con sus códigos correspondientes y se ilustran las clases más importantes.

Capítulo 3.- Validación de la solución propuesta: este capítulo presenta la descripción de la estrategia de pruebas usada, que muestran los casos de pruebas de aceptación y los resultados obtenidos.

CAPÍTULO 1: Fundamentación teórica

1.1 Introducción

En el presente capítulo se precisan elementos teóricos relacionados con estándares, especificaciones y herramientas que se aplican en la enseñanza. Se profundiza de igual forma en algunos conceptos fundamentales para la investigación, así como en la selección de lenguajes de programación web, gestores de Bases de Datos y las metodologías de desarrollo de software propuestas para el desarrollo de este trabajo.

1.2 La enseñanza virtual (e-learning)

En la sociedad del conocimiento las nuevas tecnologías de la información y las comunicaciones desempeñan un papel esencial. Como resultado de la aplicación de esas nuevas tecnologías al ámbito de la educación y de la formación surge el e-learning, que es “la utilización de las nuevas tecnologías multimediales y de Internet para mejorar la calidad del aprendizaje facilitando el acceso a recursos y servicios, así como los intercambios y la colaboración a distancia.” (AulaDiez español online)

“e-learning (electronic learning): Término que cubre un amplio grupo de aplicaciones y procesos, tales como aprendizaje basado en Web, aprendizaje basado en computadora, aulas virtuales y colaboración digital. Incluye entrega de contenidos vía Internet, intranet/extranet, audio y videgrabaciones, transmisiones satelitales, TV interactiva, CD-ROM y más.” (Guzmán, 2005)

El e-learning constituye una forma de enseñanza que presenta varias ventajas:

- Permite mejorar la calidad del aprendizaje haciendo uso de los innovadores materiales didácticos que proporcionan las TIC, facilitando la creación, adopción y distribución de contenidos.
- Permite la presentación de contenidos como textos, animaciones, gráficos, videos, y herramientas de comunicación entre alumnos y tutores como foros, chats, correo electrónico, etc.
- Permite una reducción de costos debido a reducción de gastos en viajes y un acceso a la información de forma rápida, eficaz y de buena calidad.

1.3 Objetos de aprendizaje

Un objetos de aprendizaje es “cualquier recurso digital que pueda ser reutilizado como soporte para el aprendizaje” (Wiley, 2000). También puede definirse como “un recurso con una intención formativa, compuesto de uno o varios elementos digitales, descritos con metadatos, que le posibilitan ser utilizados y reutilizados dentro de un entorno e-Learning” (Guzmán, 2005), constituyendo el elemento central en la nueva forma del diseño de cursos.

La idea de este modelo consiste en diseñar los cursos como agregados de OA, que idealmente son independientes, reutilizables y combinables como las piezas de un juego.

Los OA brindan grandes beneficios, ya que el mismo recurso puede utilizarse en distintos contextos debido a la accesibilidad, durabilidad, interoperabilidad y reutilización con que cuentan. Su control es fácil debido a la descripción dada por los metadatos y su adaptabilidad, facilitando al diseñador poder seleccionar y componer recursos según la aplicación. Su creación y reutilización se ve materializada mediante el uso de los LMS, LCMS y herramientas de autor, las cuales permiten su gestión.

1.4 Sistema de gestión del aprendizaje (LMS)

El elemento central de la comunicación en e-learning es el sistema de gestión del aprendizaje (en inglés, Learning Management System, LMS). Es un sistema basado en la web que permite el acceso a contenidos, la gestión de los recursos y la comunicación entre todos los actores implicados en el proceso (alumnos, profesores, administradores, etc.).

“La plataforma LMS permite gestionar los accesos, la actividad y permisos de los usuarios ej. (inscripción, control de qué contenidos son accedidos, notas de evaluaciones, generación de informes y estadísticas de uso, etc.) y proporciona distintas herramientas de comunicación, tanto síncronas ej. (chat o conversaciones, videoconferencia, tutorías en tiempo real, etc.) como asíncronas ej. (tablones de anuncios, foros de discusión, etc.).” (Baltasar Fernández Manjón, 2006)

Aunque los LMS poseen grandes ventajas al encargarse de todos los procesos que rodean el aprendizaje, estos a su vez carecen de facilidad a la hora de crear contenidos educativos, haciéndose necesaria la integración con un sistema especializado para la gestión de contenidos educativos (en inglés, *Learning Content Management System*, LCMS).

1.5 Sistema de gestión de contenidos educativos (LCMS)

Los sistemas de gestión de contenidos educativos o LCMS tienen su origen en los CMS (Content Management System) cuyo objetivo es simplificar la creación y la administración de los contenidos en línea. En la mayoría de los casos lo que hacen los CMS es separar los contenidos de su presentación y también facilitar un mecanismo de trabajo para la gestión de una publicación web. Los LCMS siguen el concepto básico de los CMS, que es la administración de contenidos, pero enfocados al ámbito educativo, administrando y concentrando únicamente recursos educativos y no todo tipo de información.

Los LCMS, a diferencia de los LMS, son sistemas multiusuario que están enfocados a la creación y administración de contenidos digitales, a diferentes niveles, permitiendo de esta manera reestructurar la información y los objetivos de los contenidos de manera dinámica, para crear y modificar los objetos de aprendizaje que atiendan a necesidades y estilos de aprendizaje específicos. Estos contenidos, una vez dentro del sistema pueden ser combinados, asignados a distintos cursos, descargados, etc. También existen otros tipos de herramientas especializadas en la creación de OA, a ellas se les conoce como herramientas de autor.

1.6 Herramientas de autor

El proceso de producción de un curso a distancia basado en el uso de las TIC requiere de conocimientos y habilidades que tienen su origen en el campo pedagógico e informático, por lo cual resulta bastante complejo la publicación de un curso, exigiendo del profesor cambios en su desempeño. Esta dificultad puede ser reducida con el empleo de softwares que automatizan parte del proceso, llamados herramientas de autor (HA).

Desde el punto de vista del autor los problemas referentes a la estandarización y los OAs son muy similares a otros relacionados con la introducción de las tecnologías en la educación, como las propias TICs, que en un inicio fueron catalogadas como la solución a los problemas educativos de su época. Las HA pueden ser aplicadas al desarrollo de contenidos educativos, pues permiten la creación de aplicaciones en las que, de forma sencilla y rápida se tiene la posibilidad de intercambiar el flujo de la información según las necesidades del alumno, relacionar palabras, incluir cuestionarios y marcadores que evalúen los conocimientos alcanzados, activar animaciones y vídeos explicativos, incorporar sonidos y lenguaje hablado, entre otros.

En la actualidad existe una gran variedad de HA con características similares de las cuales se describen algunas a continuación.

1.6.1 Respondus

Respondus es una poderosa herramienta para crear y administrar exámenes que se pueden imprimir en papel o ser publicados directamente a Blackboard, ANGEL, Desire2Learn, eCollege, Moodle, y otros sistemas e-learning. Los exámenes se pueden crear offline usando un entorno familiar de Windows y estos pueden ser trasladados de un sistema e-learning a otro. Las principales características de esta herramienta de autor son las siguientes:

- Incluye distintos ambientes de usuario o personalizaciones para ANGEL, Blackboard, Desire2Learn, eCollege, Moodle e IMS QTI.
- Permite crear exámenes y evaluaciones offline con una interfaz de Windows.
- Soporta hasta 15 tipos de preguntas, incluyendo cálculos y algoritmos.
- Permite importar preguntas de MS Word (incluyendo imágenes incrustadas), texto enriquecido y QTI.
- Permite insertar símbolos matemáticos y científicos usando su editor de ecuaciones interno o MathType.
- Permite añadir fácilmente imágenes, audio y video a las evaluaciones, e incluso cambiar el tamaño de las imágenes o convertirlas automáticamente. (Respondus)

1.6.2 eXeLearning

La herramienta eXeLearning es un editor XHTML que permite crear recursos multimedia interactivos sin necesidad de tener conocimientos de HTML ni de XML. Está especialmente indicado para la docencia por los módulos que incorpora: actividades de verdadero/falso, de elección múltiple, etc., así como por los archivos que nos permite incluir a la hora de desarrollar nuestro recurso (vídeos, audio mp3 o imágenes).

Una de las características principales de eXeLearning es la sencillez con la que se maneja, gracias al uso de hojas de estilo en cascada (CSS). Otra ventaja importante de eXeLearning es la utilización del estándar SCORM, lo que facilita su implementación en una plataforma virtual,

y la posibilidad que ofrece de crear un recurso abierto, de forma que el trabajo que ha realizado un profesor lo puede modificar, completar o reducir otro docente según el uso que le vaya a dar con su alumnado. (eXelearning o cómo crear recursos educativos digitales con sencillez, 2009)

1.6.3 Reload

Reload es una aplicación que ha contado con varios despliegues de herramientas, tales como Reload Editor y Reload Scorm Player, ambas de código abierto. Reload Editor es un editor de metadatos y empaquetador de contenidos, a través del cual se puede crear, importar, editar y exportar paquetes de contenido conforme a las especificaciones de ADL e IMS. Reload Learning Design Player es un reproductor de IMS LD, basado en CooperCore (reproductor de IMS LD que interpreta unidades de aprendizaje bajo IMS LD en sus tres niveles), y desarrollado en la Universidad de Bolton por Paul Sharples y Phillip Beauvoir. Construido y distribuido como aplicación de código abierto. (García, 2005)

1.6.4 CRODA

Es una aplicación Web que permite la creación de objetos de aprendizaje reutilizables, accesibles, duraderos e interoperables, de forma flexible, empleando los estándares SCORM y LOM. Es una herramienta web de fácil administración, interacción agradable y cómodo uso, que posibilita la creación de paquetes SCORM utilizados para la elaboración de cursos para los diferentes LMS. Tiene como inconveniente que no utiliza ningún estándar o especificación para la creación de los cuestionarios, imposibilitando la reutilización de los mismos en otras herramientas.

1.7 Estándares y especificaciones en e-learning

Según el diccionario de la Real Academia de la Lengua Española un estándar es lo “que sirve como tipo, modelo, norma, patrón o referencia”. En el campo técnico la estandarización es el proceso por el cual se establecen unas normas comúnmente aceptadas que permiten la cooperación de diferentes empresas o instituciones sin menoscabar su posibilidad de competir.

Existen dos tipos de estándares los oficiales o “de jure” y los “de facto”. Los estándares oficiales son aquellos que han sido aprobados y sancionados por un organismo oficial de estandarización, ya sea nacional o internacional. Los estándares de facto son aquellos que se

usan por voluntad propia o conveniencia y tienen una amplia aceptación, aunque no hayan sido sancionados por un organismo de estandarización. (Baltasar Fernández Manjón, 2006)

1.7.1 Ventajas aportadas por los estándares en e-learning

En e-learning, una de las principales funciones de los estándares es servir como facilitadores de la durabilidad y reutilización en el tiempo de los contenidos y la interoperabilidad, es decir, facilitar el intercambio de los contenidos entre diversas plataformas y sistemas.

Existen muchas ventajas asociadas a la utilización generalizada de estándares de e-learning para todas las partes implicadas en el proceso de aprendizaje. Entre ellas cabe mencionar las siguientes:

Desde el punto de vista del de los clientes o consumidores tanto institucionales como individuales, los estándares evitan quedarse atrapado por las tecnologías propietarias. Los costes se reducen al sustituir los desarrollos propios por tecnología “plug and play” de modo que, por ejemplo, una institución pueda cambiar de LMS sin tener que empezar desde el principio perdiendo toda o gran parte de la información que ya tenía en su LMS anterior.

Desde el punto de vista de los vendedores de aplicaciones, la existencia de métodos estandarizados de comunicación entre sistemas simplifica la integración de diferentes productos. Esto redundará en una reducción de los costes de desarrollo e incrementa el mercado potencial para las aplicaciones.

Desde el punto de vista de los productores de contenidos educativos, los estándares permiten que el formato de producción sea único y pueda ser utilizado en cualquier plataforma de distribución.

Desde el punto de vista de los alumnos, los estándares implican mayor posibilidad de elección del producto educativo. Además implican que los resultados de su aprendizaje (créditos o certificados) tengan mayor portabilidad. (Baltasar Fernández Manjón, 2006)

Otras ventajas y propiedades beneficiosas que se obtienen con la aplicación de los estándares:

Interoperabilidad: Que se pueda intercambiar y mezclar contenido de múltiples fuentes y se pueda usar directamente en distintos sistemas. Que sistemas diferentes puedan comunicarse, intercambiar información e interactuar de forma transparente.

Reusabilidad: Que el contenido pueda ser agrupado, desagrupado y reutilizado de forma rápida y sencilla. Que los objetos de contenido puedan ensamblarse y utilizarse en un contexto distinto a aquél para el que fueron inicialmente diseñados.

Gestionabilidad: Que el sistema pueda obtener y trazar la información adecuada sobre el usuario y el contenido.

Accesibilidad: Que un usuario pueda acceder el contenido apropiado en el momento justo y en el dispositivo correcto.

Durabilidad: Que los consumidores no queden atrapados en una tecnología propietaria de una determinada empresa. Que no haya que hacer una inversión significativa para lograr la reutilización o la interoperabilidad.

Escalabilidad: Que las tecnologías puedan configurarse para aumentar la funcionalidad de modo que se pueda dar servicio a más usuarios respondiendo a las necesidades de la institución, y que esto no exija un esfuerzo económico desproporcionado. (Baltasar Fernández Manjón, 2006)

1.8 IMS question & test interoperability specification

Los exámenes son utilizados ampliamente como herramienta para evaluar si un alumno ha asimilado los conceptos que le han sido presentados y como herramienta de autoevaluación para el mismo, de manera que éste pueda reforzar aquella parte de la materia para la que no tenga un dominio suficiente. En este sentido, el uso de herramientas informáticas ha supuesto un avance. Un gran número de tipos de preguntas pueden ser corregidas de manera automática mediante el uso de este tipo de herramientas, de manera que un profesor puede crear una batería de preguntas que el sistema informático puede utilizar para preparar exámenes y corregirlos automáticamente.

En la actualidad gran parte de las plataformas de aprendizaje incluyen con menor o mayor funcionalidad una herramienta para la creación, gestión y realización de exámenes en línea.

Sin embargo el esfuerzo realizado por el instructor en la elaboración de preguntas y exámenes puede llegar a perderse si es necesario cambiar de plataforma de aprendizaje; sería deseable tener la posibilidad de compartir este esfuerzo permitiendo el intercambio de exámenes completos, o poder crear repositorios de preguntas.

La especificación IMS Question and Test Interoperability permite crear preguntas individuales y evaluaciones completas. El objetivo principal de esta especificación es permitir el intercambio de preguntas, evaluaciones y resultados entre distintas herramientas. Con este propósito IMS QTI plantea un modelo en el que se definen los componentes principales que intervienen en el proceso de evaluación. Adicionalmente a este modelo, se proporciona un formato de contenido para almacenar las preguntas de manera independientemente del sistema o herramienta de autoría utilizada para crearlas.

Este formato, basado en XML, hace uso de estándares ampliamente utilizados en el ámbito empresarial y técnico, permitiendo el uso de las mismas preguntas entre diversos sistemas de gestión de aprendizaje o LMS, entre sistemas de evaluación electrónica independientes y la integración en un único LMS de preguntas y exámenes desarrollados con distintas herramientas. (Baltasar Fernández Manjón, 2006)

1.8.1 Versiones de la especificación

La especificación IMS QTI cuenta con cinco versiones: 1.0, 1.1, 1.2, 2.0 y 2.1. En esta investigación solo se hará alusión a las tres últimas versiones.

IMS QTI 1.2: Surge en el año 2002, y constituyó la última versión finalizada completamente en la que se abarcaron tanto preguntas individuales como exámenes completos. Una vez publicada la especificación, surgieron diversos problemas a la hora de implementarla lo que conllevó a que se publicara un suplemento (**IMS QTI 1.2.1**). Estos problemas que habían surgido requerían de grandes cambios de manera que dichas modificaciones provocarían que se perdiera la compatibilidad con las versiones anteriores. Además, algunas otras partes de la especificación necesitaban clarificarse o extenderse para resolver los problemas que habían surgido durante su puesta en práctica.

IMS QTI 2.0: Surge en el año 2005 producto al desarrollo de nuevas especificaciones como IMS Content Packaging, IMS Simple Sequencing y IMS Learning Design, para hacer compatible la especificación IMS QTI a las nuevas iniciativas.

IMS QTI 2.1: Surge en el año 2005 y está actualmente en proceso de evolución en modo borrador sobre el cual la comunidad, tanto educativa como técnica, puede opinar. El objetivo de esta nueva versión es seguir con el proceso de simplificación y evolución de la especificación, esta vez dando soporte a los exámenes completos y al intercambio de los resultados de los mismos. Además también se incluye información para clarificar la compatibilidad y el uso de IMS QTI con algunas otras de las especificaciones ya existentes. (Baltasar Fernández Manjón, 2006)

Es verídico que las especificaciones IMS aún están en desarrollo, y en el sentido estricto de la palabra todavía no son estándares, sino sólo especificaciones, sin embargo hoy permiten que los LMS empiecen a hablar entre sí, posibilitando que la portabilidad de contenidos entre distintas plataformas sea una realidad, algo que antes no era posible. (IMS Global Learning Consortium, 2011)

Después de realizar un profundo análisis de las versiones de la especificación IMS QTI se ha decidido utilizar la versión IMS QTI 2.0 por las siguientes razones. La versión 1.2 presentó problemas a la hora de su implementación dando así paso a la versión 1.2.1 la cual experimentó grandes cambios provocando la pérdida de compatibilidad con las versiones anteriores. La 2.1 aún se encuentra en proceso de evolución en modo borrador por lo cual no se considera estable, quedando así solo la versión 2.0 como la más adecuada para ser usada en este trabajo con el inconveniente de que no permite la creación de exámenes completos.

1.8.2 Perfiles de la especificación

Esta especificación define dos perfiles que se pueden utilizar como base para determinar si soportar la especificación completa o solo parte de esta. Estos perfiles son llamados QTI-Lite (que sólo se aplica al contenido) y QTI-All que conforma la especificación completa.

QTI-Lite se presenta como el perfil de nivel básico de la especificación QTI, centrándose sólo en la creación y modificación del contenido. Además no soporta los sistemas de banco de Items, ni es compatible con todas las características de la especificación, sin embargo

constituye un perfil adecuado puesto que describe la forma más sencilla de la especificación, compuesta por un subconjunto de características y funcionalidades.

Las principales diferencias entre QTI-Lite y el QTI-All son:

- QTI-Lite sólo permite una interacción por Item.
- El único tipo de interacción que soporta QTI-Lite es *choiceInteraction* y es compatible solo con las preguntas de tipo (selección simple, respuestas múltiples y verdadero/falso).
- QTI-Lite no soporta los comentarios integrados (*Feedback*).
- QTI-Lite limita los tipos de imágenes soportadas y los tipos de estructuras.
- QTI-Lite no soporta características avanzadas como items adaptativos (*adaptive items*), plantillas (*templates*) y puntuación basada en el tiempo. (IMS Global Learning Consortium, 2005)

Para el desarrollo del módulo se va a usar el perfil QTI-All pues es el que soporta los cinco tipos de preguntas que se desean implementar de las cuales dos de ellas no son soportadas por QTI-Lite.

1.8.3 Conceptos básicos de IMS QTI V 2.0

La versión dos de la especificación intenta simplificar su uso tanto desde el punto de vista técnico como desde el punto de vista del usuario de dicha especificación. Para ello, se han definido los siguientes conceptos.

1.8.3.1 Las preguntas en IMS QTI

Las preguntas individuales (*assessmentItem*) podrán ser utilizadas como un recurso educativo independiente, por ejemplo, como un recurso más dentro de un paquete IMS. Las mismas en QTI son auto-contenidas, es decir, incluyen toda la información necesaria para su presentación al alumno y su corrección automática. Toda la información relativa a la presentación ha sido agrupada en el cuerpo (*itemBody*) de las preguntas. En la presentación de la pregunta están involucrados dos aspectos:

El enunciado de la pregunta. Obviamente, la pregunta debe contener el enunciado de la misma y, de manera adicional, puede contener material explicativo complementario que permita al docente indicar el contexto en el que se realiza la pregunta. En la especificación IMS QTI v 2.X, los contenidos que podemos utilizar dentro del cuerpo de la pregunta siguen el estándar XHTML, es decir, contenido web y además es posible utilizar el estándar MathML para la representación de ecuaciones matemáticas.

La construcción de la respuesta. De manera adicional al enunciado de la pregunta, se debe dotar al alumno del equivalente del lápiz y papel para poder construir la respuesta. En el caso de IMS QTI v 2 se ha introducido el concepto de interacción (*interaction*). Dependiendo del tipo la herramienta informática generará una presentación acorde. (Baltasar Fernández Manjón, 2006)

1.8.3.2 Las interacciones en IMS QTI

En el caso de IMS QTI no está contemplado el concepto de tipo de pregunta, existiendo en su lugar el concepto de interacción. Las interacciones permiten al profesor especificar las herramientas que tendrá el alumno disponible para poder construir la respuesta.

Al igual que existen múltiples tipos de preguntas, también existen múltiples tipos de interacción. A continuación se describirán los tipos de interacciones posibles que pueden utilizarse dentro de una pregunta.

Hay que remarcar que existen dos grupos de interacciones, las interacciones en línea y las interacciones en bloque. Las interacciones en línea son un tipo de interacción que pueden incluirse en medio del enunciado de la pregunta. Por otra parte las interacciones de tipo bloque están pensadas para ser presentadas de manera independiente al enunciado de la pregunta. (Baltasar Fernández Manjón, 2006)

Para ejemplificar las posibles interacciones debido a la falta de herramientas que brinden un soporte completo a la especificación, se harán uso de los ejemplos que proporciona la propia especificación.

Interacciones Simples

Las interacciones simples son aquellas interacciones en las que la corrección de las mismas se realiza en base a la selección de una opción o varias opciones disponibles.

Las interacciones que pertenecen a esta categoría son:

choiceInteraction: Esta interacción muestra al alumno un conjunto de posibles opciones. El alumno podrá seleccionar una o varias posibles opciones como respuesta. Es posible indicar que el conjunto de posibles opciones sea barajado.

UNATTENDED LUGGAGE

Look at the text in the picture.

**NEVER LEAVE
LUGGAGE
UNATTENDED**

What does it say?

| | |
|---|-----------------------|
| You must stay with your luggage at all times. | <input type="radio"/> |
| Do not let someone else look after your luggage. | <input type="radio"/> |
| Remember your luggage when you leave. | <input type="radio"/> |

Figura 1.1. Ejemplo de choiceInteraction (selección simple)
Fuente: (IMS Global Learning Consortium, 2005)

COMPOSITION OF WATER

Which of the following elements are used to form water?

| | |
|-----------------|-------------------------------------|
| Carbon | <input type="checkbox"/> |
| Oxygen | <input checked="" type="checkbox"/> |
| Hydrogen | <input checked="" type="checkbox"/> |
| Chlorine | <input type="checkbox"/> |
| Helium | <input type="checkbox"/> |
| Nitrogen | <input type="checkbox"/> |

Figura 1.2. Ejemplo de choiceInteraction (respuestas múltiple)

Fuente: (IMS Global Learning Consortium, 2005)

orderInteraction: En esta interacción el objetivo del alumno es reordenar el conjunto de soluciones proporcionada. Además, es posible un número mínimo y un número máximo de opciones que conforman la solución, de manera que se realizaría una selección sobre las opciones disponibles y posteriormente se realizaría una ordenación de los elementos de dicha selección.

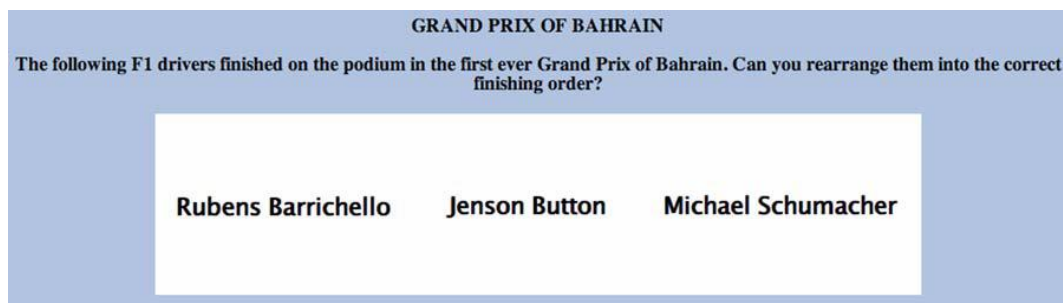
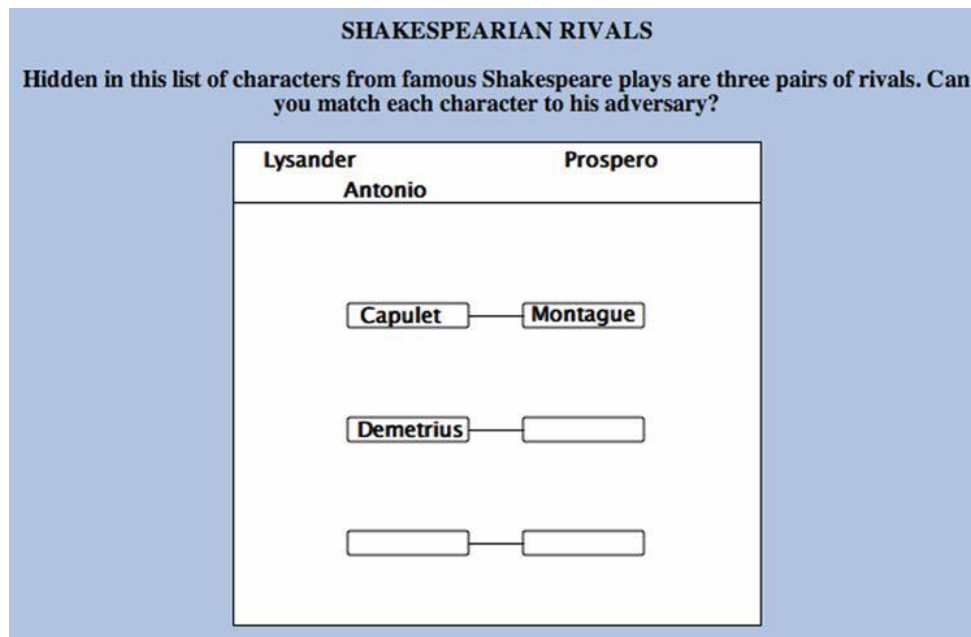


Figura 1.3. Ejemplo de orderInteraction.

Fuente: (IMS Global Learning Consortium, 2005)

associateInteraction: Esta interacción presenta al alumno un conjunto de opciones y permite crear asociaciones por parejas entre dichas opciones. Es posible indicar el número mínimo y máximo de asociaciones que deben crearse como parte de la respuesta. Además, también es posible indicar el número mínimo y máximo de veces que una de las opciones puede aparecer dentro de una asociación.



*Figura 1.4. Ejemplo de associateInteraction.
Fuente: (IMS Global Learning Consortium, 2005)*

matchInteraction: Esta interacción presenta al alumno dos conjuntos de opciones y le permite crear pares de asociaciones entre ellas. Al igual que en la interacción anterior es posible indicar el número mínimo y máximo de asociaciones posibles o el número mínimo y máximo de apariciones de una de las opciones en las asociaciones creadas.

CHARACTERS AND PLAYS

| Match the following characters to the Shakespeare play they appeared in: | The Tempest | Romeo and Juliet | A Midsummer-Night's Dream |
|--|-----------------------|-----------------------|---------------------------|
| Prospero | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Capulet | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Demetrius | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Lysander | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

*Figura 1.5. Ejemplo de matchInteraction (enlazar columna).
Fuente: (IMS Global Learning Consortium, 2005)*

gapMatchInteraction: Esta interacción permite definir un conjunto de huecos dentro del enunciado de la pregunta a mostrar al alumno. Además se permitirá al alumno asociar a cada uno de los huecos una de las posibles opciones de respuesta. Hay que destacar que las opciones posibles son compartidas por todos los huecos. Como posibles respuestas es posible utilizar texto o también es posible utilizar imágenes. Además es posible restringir el número

mínimo y máximo de veces que es utilizada cada una de las posibles opciones del conjunto de respuestas. (Baltasar Fernández Manjón, 2006)

RICHARD III (TAKE 1)

Identify the missing words in this famous quotation from Shakespeare's Richard III.

Now is the of our discontent
Made glorious by this sun of York;
And all the clouds that lour'd upon our house
In the deep bosom of the ocean buried.

Use the table below to select the missing words.

| | winter | spring | summer | autumn |
|--------|----------------------------------|-----------------------|-----------------------|-----------------------|
| Word 1 | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| Word 2 | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

*Figura 1.6. Ejemplo de gapMatchInteraction.
Fuente: (IMS Global Learning Consortium, 2005)*

Interacciones de Texto

En esta categoría se encuentran las interacciones en las que la respuesta que construirá el alumno puede ser una única palabra, una frase corta o un párrafo de texto completo. Estas interacciones permiten que durante el proceso de corrección se tenga en cuenta la respuesta en forma de texto que ha construido el alumno. Las interacciones que pertenecen a esta categoría son:

inlineChoiceInteraction (interacción en línea): Esta interacción está pensada para definir un hueco donde se permitirá al alumno escoger entre un conjunto de opciones, donde cada una de estas opciones una palabra o frase corta. A diferencia de *gapMatchInteraction*, esta interacción está ideada para que cada uno de los huecos pueda tener un conjunto de opciones independiente.

RICHARD III (TAKE 2)

Identify the missing word in this famous quotation from Shakespeare's Richard III.

Now is the winter of our discontent
Made glorious summer by this sun of ;
And all the clouds that lour'd upon our house
In the deep bosom of the ocean buried.

*Figura 1.7. Ejemplo de inlineChoiceInteraction.
Fuente: (IMS Global Learning Consortium, 2005)*

textEntryInteraction (interacción en línea): Al igual que la interacción anterior, esta interacción tiene como objetivo crear un hueco donde se permitirá teclear una palabra o frase corta para poder construir la respuesta. Cuando se define una pregunta con esta interacción es posible especificar la longitud del texto que se espera que el alumno introduzca.

RICHARD III (TAKE 3)

Identify the missing word in this famous quotation from Shakespeare's Richard III.

Now is the winter of our discontent
Made glorious summer by this sun of ;
And all the clouds that lour'd upon our house
In the deep bosom of the ocean buried.

*Figura 1.8. Ejemplo de textEntryInteraction (espacios en blanco).
Fuente: (IMS Global Learning Consortium, 2005)*

extendedTextInteraction: Esta interacción está pensada para que el alumno construya como respuesta un párrafo de texto. Es posible indicar el número mínimo y máximo de líneas de texto esperadas, junto con la longitud máxima de cada una de ellas.

WRITING A POSTCARD

Read this postcard from your English pen-friend, Sam.

Here is a postcard of my town. Please send me a postcard from your town. What size is your town? What is the nicest part of your town? Where do you go in the evenings?

Sam.

Write Sam a postcard. Answer the questions. Write 25-35 words.

*Figura 1.9. Ejemplo de extendedTextInteraction.
Fuente: (IMS Global Learning Consortium, 2005)*

hottextInteraction: El objetivo de esta interacción es que el alumno seleccione partes de texto que estarán resaltadas en el enunciado de la pregunta. Es posible indicar el número mínimo y máximo de elecciones que debe realizar el alumno, siendo 0, el valor mínimo, y 1, el valor máximo, de selecciones por defecto. (Baltasar Fernández Manjón, 2006)

IDENTIFYING SENTENCE ERRORS

Select the error in the following passage of text (or *No Error* if there is none).

Sponsors of the Olympic Games **who bought** advertising time on United States television **includes** **at least** a dozen international firms **whose** names are familiar to American consumers. **No error.**

*Figura 1.10. Ejemplo de hottextInteraction.
Fuente: (IMS Global Learning Consortium, 2005)*

La especificación cuenta con más interacciones como las gráficas, entre otras, las cuales no serán tratadas en el presente trabajo, porque los cinco tipos de preguntas contemplados en el módulo, están contenidos dentro de las interacciones simples y de texto.

A continuación se muestran los tipos de interacciones que soportan las preguntas a implementar en el módulo:

- choiceInteraction (verdadero o falso, selección simple y respuestas múltiples).
- matchInteraction (enlazar columna).
- textEntryInteraction (espacios en blanco).

1.8.4 Herramientas relacionadas con la especificación IMS QTI

Existen diversas herramientas que proporcionan soporte en mayor o menor medida a la especificación IMS QTI en sus diferentes versiones. Las herramientas pueden ser divididas en tres categorías:

- **Herramientas de Autoría.** Herramientas que permitirán crear exámenes y finalmente salvarlos en el formato IMS QTI.
- **LMS.** En este caso los LMS incluirán una herramienta de gestión de evaluaciones en línea donde la herramienta permitirá importar y/o exportar las preguntas y exámenes en formato IMS QTI.
- **Reproductores y motores de ejecución.** Los reproductores interpretarán preguntas y exámenes compatibles con IMS QTI y permitirán realizar el examen. Los motores de ejecución, proporcionan infraestructura y varios componentes para la creación de herramientas de evaluación en línea, en este caso compatibles con IMS QTI.

Algunas herramientas de autoría son:

- Respondus
- QuestionMark's Perception
- Assesst Designer
- AQuRate
- Mathqurate

Algunos LMS que proporcionan soporte para IMS QTI:

- Alphastudy

- ANGEL Learning Management Suite
- DB Primary
- ILIAS
- Claroline
- SAKAI
- .LRN
- OLAT
- WebCT

Algunos LCMS que proporcionan soporte para IMS QTI:

- Chamilo
- Dokeos
- ATutor
- Moodle
- Online Learning And Training

Sistemas que permiten la visualización y ejecución de QTI:

- <e-QTI>
- Q-Player
- APIS
- R2Q2
- MathQTI

1.9 Herramientas y tecnologías a utilizar

Para el desarrollo de este módulo se deben de tener en cuenta una serie de tecnologías y herramientas imprescindibles para su realización. Para lograr esta correcta selección se deben estudiar cada una de las existentes analizando sus características y ventajas además de mantener las políticas de desarrollo del centro.

1.9.1 Metodología de desarrollo de software

El desarrollo de software resulta verdaderamente complicado. El mismo engloba una serie de bases, herramientas y métodos que necesitan imperativamente de una metodología, con el objetivo de encausar a cada uno de estos engranajes en lo que sería un proceso coordinado y acorde al fin que se persigue.

Hoy en día existen diferentes propuestas metodológicas con el fin de alcanzar una mayor planificación, mejor costo y ganancia de tiempo, dentro de estas se encuentran las metodologías ágiles como extreme programming.

1.9.1.1 Extreme programming (XP)

Esta metodología de desarrollo de software se caracteriza por su simplicidad. Posibilita simplificar el diseño para agilizar el desarrollo y facilitar el mantenimiento. Permite un diseño complejo del código junto a sucesivas modificaciones por parte de diferentes desarrolladores haciendo que la complejidad aumente exponencialmente. La metodología consiste en una programación rápida o extrema, cuya particularidad es tener como parte del equipo, al usuario final, pues es uno de los requisitos para llegar al éxito del proyecto. Se inserta dentro de la familia de metodologías ligeras, tratando de obtener métodos sencillos de obtener software de calidad. Uno de sus principios básicos es la mejora de la comunicación con los usuarios y los desarrolladores, la simplicidad, al desarrollar y codificar los módulos del sistema. También existen otros tipos de metodologías como las pesadas dentro de la cual se encuentra el proceso unificado de desarrollo o RUP.

1.9.1.2 Proceso unificado de desarrollo (RUP)

El proceso unificado de desarrollo constituye una metodología pesada, dirigida a grandes proyectos, está centrado en la arquitectura, guiado por casos de uso y es iterativo e incremental.

Define como sus principales elementos:

- Trabajadores (“quién”): Define el comportamiento y responsabilidades (rol) de un individuo, grupo de individuos, sistema automatizado o máquina, que trabajan en conjunto con un equipo.
- Actividades (“cómo”): Es una tarea que tiene un propósito claro, es realizada por un trabajador y manipula elementos.
- Artefactos (“qué”): Productos tangibles del proyecto que son producidos, modificados y usados por las actividades. Pueden ser modelos, elementos dentro del modelo, código fuente y ejecutables.
- Flujo de Actividades (“cuándo”): Secuencia de actividades realizadas por los trabajadores y que produce un resultado de valor observable.

Las actividades son agrupadas en grupos lógicos, donde se definen 9 flujos de trabajo fundamentales, de los cuales los 6 primeros son considerados flujos de ingeniería y los 3 restantes como flujos de apoyo. Estos flujos son: Modelado de Negocio, Requerimientos, Análisis y Diseño, Implementación, Pruebas, Despliegue, Configuración y Administración de Cambios, Administración de Proyecto y Entorno.

En el desarrollo del módulo se va a hacer uso de esta metodología por las características de robustez y generación de gran cantidad de información, permitiendo plasmar toda la información necesaria del módulo, además por política del proyecto es la metodología que se utiliza.

1.9.2 Tecnologías a utilizar en la Web

Los lenguajes pueden dividirse en dos grupos, los lenguajes del lado del Servidor y los lenguajes del lado del Cliente, donde ambos reconocen la filosofía de la arquitectura Cliente/Servidor para las plataformas Web. En el lado del cliente existen tecnologías como: JavaScript además de Cascading Style Sheets (CSS) y HyperText Markup Language (HTML), destinados a ofrecer dinamismo a la aplicación donde se halla utilizado contribuyendo a la no recarga en servidores, proporcionando rapidez y optimización en los canales de comunicación. Aunque también existen técnicas de desarrollo para la web como lo es Ajax y XML.

Asynchronous JavaScript (Ajax)

Ajax está conformado con el objetivo de crear aplicaciones interactivas o Rich Internet Applications (RIA), es una tecnología asíncrona, en el sentido que los datos adicionales se cargan en segundo plano sin interferir con la visualización ni el comportamiento de la página. Además de ser una técnica válida para múltiples plataformas y utilizable en muchos sistemas operativos y navegadores dado que está basado en estándares abiertos como JavaScript y Document Object Model (DOM).

Javascript (JScript)

Es un lenguaje de programación interpretado con sintaxis semejante a Java y C, al igual que Java, JavaScript es orientado a objetos propiamente dicho, pues dispone de herencia, esta se realiza siguiendo el paradigma de programación basada en prototipos, debido a que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad. Todos los navegadores modernos interpretan el código JavaScript integrado dentro de las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del DOM.

Extensible Markup Language (XML)

De sus siglas en inglés Extensible Markup Language (XML), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium (W3C). Es una simplificación y adaptación del SGML y permite definir la gramática de lenguajes. Por lo tanto, XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades.

El XML permite crear etiquetas adaptadas a las necesidades y es especialmente estricto en cuanto a lo que está permitido y lo que no, todo documento debe cumplir dos condiciones: ser válido y estar bien formado.

Este metalenguaje será usado en el presente trabajo para proporcionar un formato de contenido para almacenar las preguntas de manera independientemente del sistema o herramienta de autoría utilizada para crearlas.

Cascading Style Sheets (CSS)

Las hojas de estilo en cascada (Cascading Style Sheets) son un lenguaje formal usado para definir la presentación estética de un documento estructurado y escrito en HTML. La idea que se encuentra detrás del desarrollo de CSS es separar la estructura y el contenido de la presentación estética en un documento. Esto permite un control mayor del documento y sus atributos convirtiendo al HTML en un documento versátil y liviano.

HyperText Markup Language (HTML)

Se traduce al español como lenguaje de etiquetas de hipertexto. Es el lenguaje de marcado predominante para la construcción de páginas web, diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas Web. Permite además códigos de lenguajes de programación extendiendo su capacidad y funcionalidad. Es el estándar usado en el World Wide Web. HTML ofrece los medios a través de los cuales se publica online documentos con cabeceras, texto, fotos, tablas además de incluir hojas de cálculo y otras aplicaciones directamente en los documentos.

1.9.3 Procesador de hipertexto (PHP)

PHP es un lenguaje de programación interpretado del lado del servidor gratuito e independiente de la plataforma, rápido, con una gran librería de funciones y mucha documentación. Este puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno.

Dentro de las ventajas que supone el uso de PHP se destacan las siguientes:

- Es un lenguaje multiplataforma que soporta la programación orientada a objetos y puede ser utilizado tanto sobre Linux como Windows.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos.
- Presenta una capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destacándose su conectividad con MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras.
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.

- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución.

1.9.4 Framework

A partir del desarrollo alcanzado en el mundo de las tecnologías ha sido necesaria la creación de frameworks con el objetivo de facilitar el desarrollo del software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los detalles de bajo nivel para proveer un sistema funcional.

En el desarrollo de software, el framework constituye la estructura conceptual y tecnológica que está definida con artefactos o módulos de software concretos, en base a la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros programas para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Los frameworks simplifican el desarrollo de las aplicaciones mediante la automatización de muchas de las tareas comunes. Además el framework proporciona una estructura al código fuente, forzando al programador a crear un código más legible y fácil de mantener.

1.9.4.1 Symfony

Symfony es uno de los frameworks PHP más populares entre los usuarios y las empresas, ya que automatiza las tareas más comunes, permitiendo a los desarrolladores dedicarse por completo a los aspectos específicos de cada aplicación. Este separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Symfony es maduro, estable, profesional y está muy bien documentado. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web.

Symfony está desarrollado completamente con PHP5. Es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y Microsoft SQL Server. Se

puede ejecutar tanto en plataformas Unix (Unix, Linux, etc.) como en plataformas Windows. Además, utiliza el Modelo Vista Controlador como patrón de diseño web. (Potencier)

Modelo vista controlador (MVC)

El Modelo Vista Controlador es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el sistema de gestión de base de datos y la lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.

Al incorporar el modelo de arquitectura MVC a un diseño, las piezas de un programa se pueden construir por separado y luego unir las en tiempo de ejecución. Si uno de los componentes, posteriormente, se observa que funciona mal, puede reemplazarse sin que las otras piezas se vean afectadas.

Dentro de las ventajas que supone el uso de este estilo arquitectónico se encuentran:

- Presenta soporte de vistas múltiples: dado que la vista se halla separada del modelo y no hay dependencia directa del modelo con respecto a la vista, la interfaz de usuario puede mostrar múltiples vistas de los mismos datos simultáneamente.
- Adaptación al cambio: los requerimientos de interfaz de usuario tienden a cambiar con mayor rapidez que las reglas de negocios. Dado que el modelo no depende de las vistas, agregar nuevas opciones de presentación generalmente no afecta al modelo.

1.9.4.2 ExtJS

ExtJS es una librería JavaScript ligera y de alto rendimiento destinada a crear entornos de aplicaciones web complejos compatible con la mayoría de los navegadores, que permiten crear aplicaciones enriquecidas del lado del cliente usando tecnologías como AJAX, DHTML y DOM. Fue creado por Jack Slocum originalmente como extensión para Yahoo User Interface (YUI), en la actualidad puede usarse como extensión para las bibliotecas jQuery y Prototype. (Shea Frederick, 2008)

Esta librería incluye:

- Componentes UI del alto performance y personalizables.
- Modelo de componentes extensibles.
- Un API fácil de usar.
- Licencias de código abierto y comercial.

1.9.5 Sistema gestor de base de datos (SGBD)

Un sistema de gestión de base de datos es un software que funciona como interfaz entre el usuario, la aplicación y la base de datos. Tiene como propósito el manejo de datos y entidades de forma sencilla dentro de la base de datos. Con el uso de los gestores de bases de datos se logra una independencia de los datos y los programas de aplicación, la seguridad y protección de los datos, integración y sincronización de las bases de datos, integridad, facilidad de manipulación de la información y control centralizado.

1.9.5.1 Oracle

Oracle es un sistema de gestión de base de datos objeto-relacional, desarrollado por Oracle Corporation. Se considera a Oracle como uno de los sistemas de bases de datos más completos. Es un producto vendido a nivel mundial, aunque la gran potencia que tiene y su elevado precio hacen que sólo se vea en empresas muy grandes y multinacionales. Entre sus características principales cabe destacar el soporte de transacciones, estabilidad, escalabilidad y soporte multiplataforma. En el desarrollo de páginas web no es muy usado porque es un sistema muy caro que no está tan extendido como otras bases de datos, por ejemplo, Postgres, MySQL, SQL Server, etc.

1.9.5.2 MySQL

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario. Por un lado se ofrece bajo la Licencia Pública General de GNU (GPL) para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Es uno de los SGBD más utilizados en el mundo, y es muy popular por estar ampliamente ligado a PHP. Funciona sobre múltiples plataformas, incluyendo GNU/Linux, Mac OS X, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista, Windows 7 y otras versiones de Windows.

1.9.5.3 PostgreSQL

Entre los sistemas de bases de datos existentes hoy en día, PostgreSQL juega un papel muy importante ya que es un sistema que tiene muchas cualidades que lo hacen ser una muy buena alternativa para instalar sistemas en empresas, universidades y una gran cantidad de otras aplicaciones. Es un avanzado sistema de bases de datos relacionales basado en código abierto bajo licencia BSD.

PostgreSQL se caracteriza por ser un sistema estable, de alto rendimiento y de gran flexibilidad ya que funciona sobre la mayoría de los sistemas Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows. Este cuenta una gran comunidad de profesionales y empresas que ofrecen soporte, de la cual el grupo global de desarrollo de PostgreSQL puede obtener beneficios y contribuir en su mejoramiento.

Indiscutiblemente PostgreSQL es uno de los mejores sistemas de bases de datos, presenta numerosas ventajas, es robusto, estable y actualmente en nuestra universidad se ha generalizado su uso. Por todos los argumentos antes mencionados se estima conveniente el uso de este sistema en el módulo.

1.9.6 Servidor web

Un servidor web es un software en un servidor que está diseñado para transferir hipertextos, páginas web o páginas HTML, además se encuentra en constante procesamiento al mantenerse en espera de solicitudes de ordenadores clientes a través de navegadores web y dándole respuesta a estas mediante páginas web que se muestran en dicho navegador. Existen varios tipos de servidores como: los servidores de correo, de chat, de fax y de Web.

1.9.6.1 Zope

Es un servidor de aplicaciones Web, pertenece a la gran familia de software libre, es desarrollado en Python. Entre las características que lo distinguen se encuentra su base de datos orientada a objetos denominada Zope Object Database (ZODB), la cual almacena objetos ordenados en un sistema similar a un sistema de ficheros, pero cada objeto tiene propiedades, métodos u otros objetos. Esta aproximación difiere bastante de las bases de datos relacionales actuales. No obstante, Zope posee la ventaja de contar con múltiples

conectores para las diferentes bases de datos relacionales ofreciendo sistemas básicos de conexión y consulta abstrayéndolos como objetos.

1.9.6.2 Internet information services (IIS)

Dentro de la lista de servidores web destacados, cabe mencionar a Internet Information Services el cual ofrece un conjunto de servicios como: File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), Network News Transport Protocol (NNTP) y HTTP/HTTPS para ordenadores con el sistema operativo Microsoft Windows, por lo que es propiedad de Microsoft Corporation. Se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas como Active Server Pages (ASP), ASP.NET y también pueden ser incluidos los de otros fabricantes, como PHP o Perl.

1.9.6.3 Apache

Uno de los servidores Web más utilizados a nivel mundial es el servidor Apache. Es un servidor web HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras. Su diseño le permite ser un servidor web potente y flexible que funciona en una amplia gama de plataformas y entornos. Este servidor es funcional tanto para páginas estáticas como para páginas dinámicas a través de otras herramientas soportadas que facilitan la actualización de los contenidos usando bases de datos, ficheros u otras fuentes de información.

Aunque existen varios servidores web de muy buena calidad y rendimiento que constituyen una buena opción a utilizar, cabe destacar que Apache posee algunas ventajas que lo hacen realmente exclusivo para su selección en este proyecto, debido a su: estructuración en módulos, gran compatibilidad con el lenguaje de programación PHP y la experiencia acumulada sobre su uso tanto a nivel mundial como en la Universidad de las Ciencias Informáticas.

1.9.7 Entorno de desarrollo integrado (IDE)

Un entorno de desarrollo integrado (en inglés Integrated Development Environment o IDE) es un programa compuesto por una serie de herramientas que utilizan los programadores para desarrollar código. Esta herramienta puede estar pensada para su utilización con un único lenguaje de programación o bien puede dar cabida a varios de estos.

Las herramientas que normalmente componen un entorno de desarrollo integrado son las siguientes: un editor de texto, un compilador, un intérprete, unas herramientas para la automatización, un depurador, un sistema de ayuda para la construcción de interfaces gráficas de usuario y opcionalmente un sistema de control de versiones.

Hoy en día los entornos de desarrollo proporcionan un marco de trabajo para la mayoría de los lenguajes de programación existentes en el mercado (por ejemplo C, C++, C#, Java, Python, Visual Basic y PHP entre otros). Además es posible que un mismo entorno de desarrollo tenga la posibilidad de utilizar varios lenguajes de programación, como es el caso de Eclipse.

A continuación se presentan algunos de ellos con el objetivo de mostrar sus principales características en función de seleccionar el más indicado para realizar la implementación.

1.9.7.1 Eclipse

Eclipse es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar aplicaciones. Constituye un armazón sobre el que se pueden montar herramientas de desarrollo para cualquier lenguaje, mediante la instalación de los plugins adecuados. Permite integrar diversos lenguajes sobre un mismo IDE e introducir otras aplicaciones y accesorios que pueden resultar útiles durante el proceso de desarrollo como: herramientas UML, editores visuales de interfaces y ayuda en línea para librerías. Eclipse también ha demostrado su calidad en el desarrollo de aplicaciones web mediante el plugin para el lenguaje PHP. Dicho plugin permite a los desarrolladores en PHP realizar un trabajo rápido, cómodo y sencillo pues tiene auto completamiento y colorea los errores.

1.9.7.2 NetBeans

NetBeans es un entorno de desarrollo libre, gratuito y sin restricciones de uso, hecho principalmente para el lenguaje de programación Java. Existe además un número importante de módulos para extender el NetBeans IDE. Es una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java pero puede servir para cualquier otro lenguaje de programación, como PHP5 y soporta el desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles).

NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento, y con cerca de 100 socios en todo el mundo. Sun

MicroSystems fundó el proyecto de código abierto NetBeans en junio de 2000 y actualmente su propietario es Oracle.

Teniendo en cuenta cada una de las características que hacen que tanto NetBeans como el Eclipse posean verdaderas ventajas para la implementación de este proyecto se selecciona a NetBeans, debido a que actualmente el mismo cuenta con un constante crecimiento y actualización. Además el equipo de trabajo cuenta con mayor experiencia en su uso.

1.10 Conclusiones

Este capítulo constituye la base fundamental sobre la cual se construye la solución del presente problema, abordándose los conceptos y definiciones fundamentales que se basan en el e-learning y toman parte en el desarrollo de este trabajo. Además se justificó la utilización de las principales aplicaciones, tecnologías y estándares que posibilitan este proceso de desarrollo.

CAPÍTULO 2: Implementación

2.1 Introducción

En este capítulo se describe la implementación llevada a cabo en el módulo ejercicios, así como la estructura del sistema mediante los diagramas de paquetes, la organización de los componentes creados, la estructura de la base de datos relacional y la adopción de un estándar de codificación.

2.2 Requisitos funcionales

Los requisitos funcionales (RF) definen el comportamiento interno del software como los detalles técnicos, la manipulación de datos y otras funcionalidades específicas que muestran cómo los casos de uso serán llevados a la práctica. A continuación se describen los RF correspondientes a la presente aplicación.

El módulo deberá permitir la creación de cuestionarios contenidos por preguntas bajo el estándar IMS QTI v2.0 en formato XML.

Dentro del estándar IMS QTI solo se van a contemplar cinco tipos de preguntas:

- Verdadero o falso: preguntas para contestar verdadero o falso.
- Selección simple: seleccionar una respuesta entre varias posibilidades.
- Respuestas múltiples: seleccionar varias respuestas correctas entre múltiples posibilidades.
- Enlazar columna: se presentan dos conjuntos de opciones que permiten crear pares de asociaciones entre ellos.
- Espacios en blanco: rellenar con texto espacios en blanco dentro de un texto dado.

Datos generales requeridos por las preguntas:

- Nombre o título de la pregunta: Es un campo obligatorio que permite reconocer la pregunta en un listado.

- Texto de la pregunta: Constituye el texto que se le muestra al alumno como el enunciado de la pregunta, el cual puede contener imágenes, fórmulas, etc.
- Nota de la pregunta: Es un valor numérico obligatorio que representa la nota de la pregunta.
- Retroalimentación general: Es la retroalimentación que se muestra siempre al terminar una pregunta.

Datos específicos requeridos por preguntas:

Verdadero o falso

- Respuesta correcta: Es un campo que solo debe permitir seleccionar dos opciones (Verdadero o Falso).
- Retroalimentación para la respuesta verdadera: Retroalimentación que se muestra en caso de estar correcta la respuesta.
- Retroalimentación para la respuesta falsa: Retroalimentación que se muestra en caso de estar incorrecta la respuesta.

Selección simple y respuestas múltiples:

Para estos tipos de preguntas se necesita introducir un conjunto de opciones con los siguientes campos:

- Respuesta: Es la respuesta que se muestra como posible opción a seleccionar.
- Peso: Constituye el valor que tiene la opción dentro de la pregunta.
- Retroalimentación: Es la retroalimentación que el sistema debe mostrar en caso de que dicha opción sea marcada.

Enlazar columna:

Esta pregunta requiere de un conjunto de opciones con los siguientes campos.

- Respuesta: Es la respuesta de esta opción.

- Peso: Constituye el valor que tiene la opción dentro de la pregunta.
- Pregunta: Es la pregunta de esta opción.

Espacios en blanco:

Esta pregunta solo requiere de un campo para introducir el texto de la pregunta. Para dejar un espacio en blanco se deben de usar corchetes.

El sistema debe permitir las siguientes funcionalidades:

Añadir y eliminar cuestionarios.

Modificar los datos de los cuestionarios.

Añadir preguntas a los cuestionarios, de todos los tipos contemplados por la aplicación.

Reutilizar preguntas asignadas a un cuestionario en otro.

Desvincular una pregunta de un cuestionario.

Eliminar preguntas de los cuestionarios.

Crear, eliminar y modificar preguntas.

Exportar e importar cuestionarios bajo el estándar IMS QTI v2.0 en formato XML.

Exportar e importar preguntas bajo el estándar IMS QTI v2.0 en formato XML.

Mostrar una previsualización de los cuestionarios y preguntas para comprobar que se han creado de forma correcta.

Validar que los datos de cada campo sean validados por el lado del cliente y del servidor.

2.3 Modelo de implementación

El modelo de implementación es una colección de componentes y los subsistemas que los contienen que describen cómo los elementos del modelo de diseño se implementan en términos de componentes. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para el despliegue del sistema. Este

modelo describe también cómo se organizan los componentes de acuerdo con los mecanismos de estructuración y modularización disponibles en el entorno de implementación y en el lenguaje o lenguajes de programación utilizados y cómo dependen los componentes unos de otros.

2.3.1 Diagrama de paquetes

En la figura 2.1 se muestra el diagrama de paquetes general en el cual se pueden observar las dos partes en que se encuentra dividido el sistema: por un lado podemos apreciar la parte referente al framework symfony en su versión 1.4 y por el otro la librería de javascript EXT JS 3.2.1.

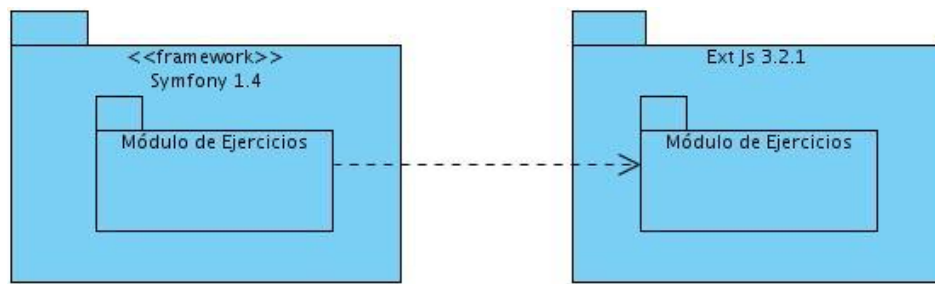


Figura 2.1. Diagrama de paquetes general del sistema.

En la versión 2.0 de la herramienta de autor CRODA se creó una estructura estandarizada y de esta forma extrapolar la forma de estructurar los archivos en las versiones más recientes del symfony y aplicarlo a la estructura de la librería javascript EXT JS.

Con el propósito de lograr lo anteriormente planteado se crearon nuevos scripts que se integraron al resto de las dependencias de la librería siguiendo el patrón del framework. En las carpetas apps, extjs y core se agruparon los archivos siendo esta última la contenedora de los scripts: core.js y module.js. Ver figura 2.2

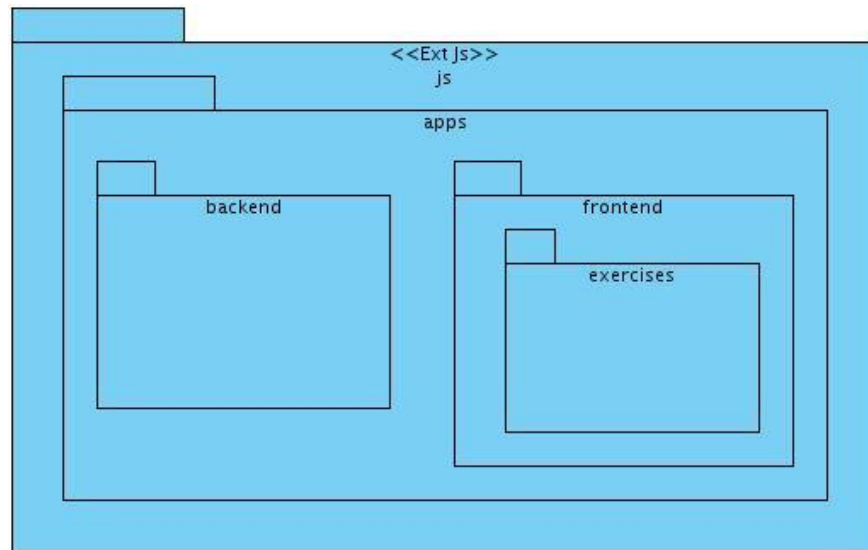


Figura 2.2. Diagrama de paquetes de EXT JS.

Los scripts mencionados anteriormente son los que permiten a las librerías del EXT JS funcionar como si se tratase del propio symfony sobre sus diferentes módulos.

Gracias a esta forma de organización fue posible tratar los conjuntos de scripts referentes a un módulo en el symfony como módulos independientes dentro del EXT JS. Los mismos se encuentran dentro de la carpeta apps, y es por ello que el conjunto de componentes, funciones y scripts relacionadas con el módulo de ejercicios se encuentran dentro de esta localización en la carpeta exercises de forma ordenada.

2.3.2 Diagrama de componentes

El flujo de información en la aplicación web inicia en el index.php de framework, el cual redirecciona sobre el actions.class.php del módulo main. Esta clase tiene asociada una plantilla (*template*) que a través del *layout.php* se encarga de dar paso a un parcial (partial) denominado: *_left_menu.php*. Este parcial se ocupa del lateral izquierdo de la página de inicio y de hacer la petición al script *modMain.js* de la librería EXT JS, el cual maneja el área de los componentes visuales en dicha librería (*fronted*). Ver figura 2.3.

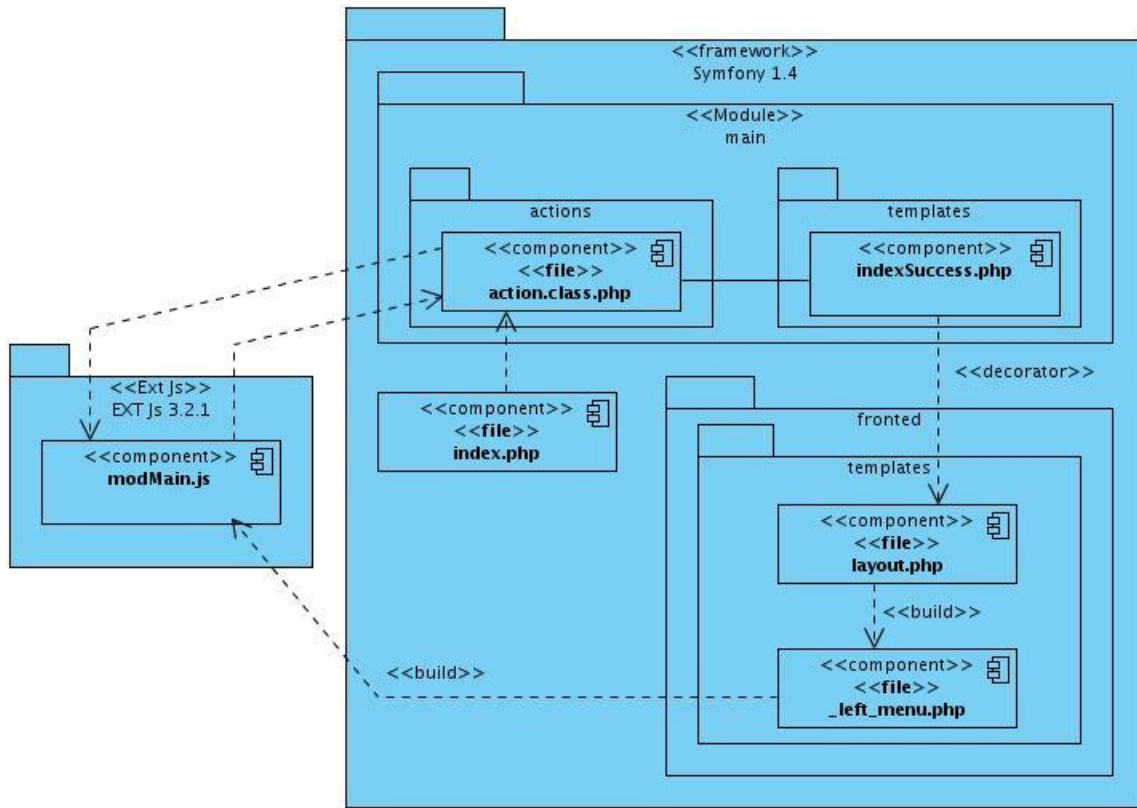


Figura 2.3. Diagrama de componentes general del sistema.

En el modMain.js se encuentra la función buildTreeCreation la cual al ser invocada envía una petición a la acción (action) del módulo principal (main) y este recibe lo que debe graficar en el árbol de opciones que despliega.

A partir del script modMain se cargan los principales componentes del módulo ejercicios, modExercise.js, cmpGeneralView.js y cmpTreeExercise.js que son los encargados de manejar la interfaz principal. Ver figura 2.4.

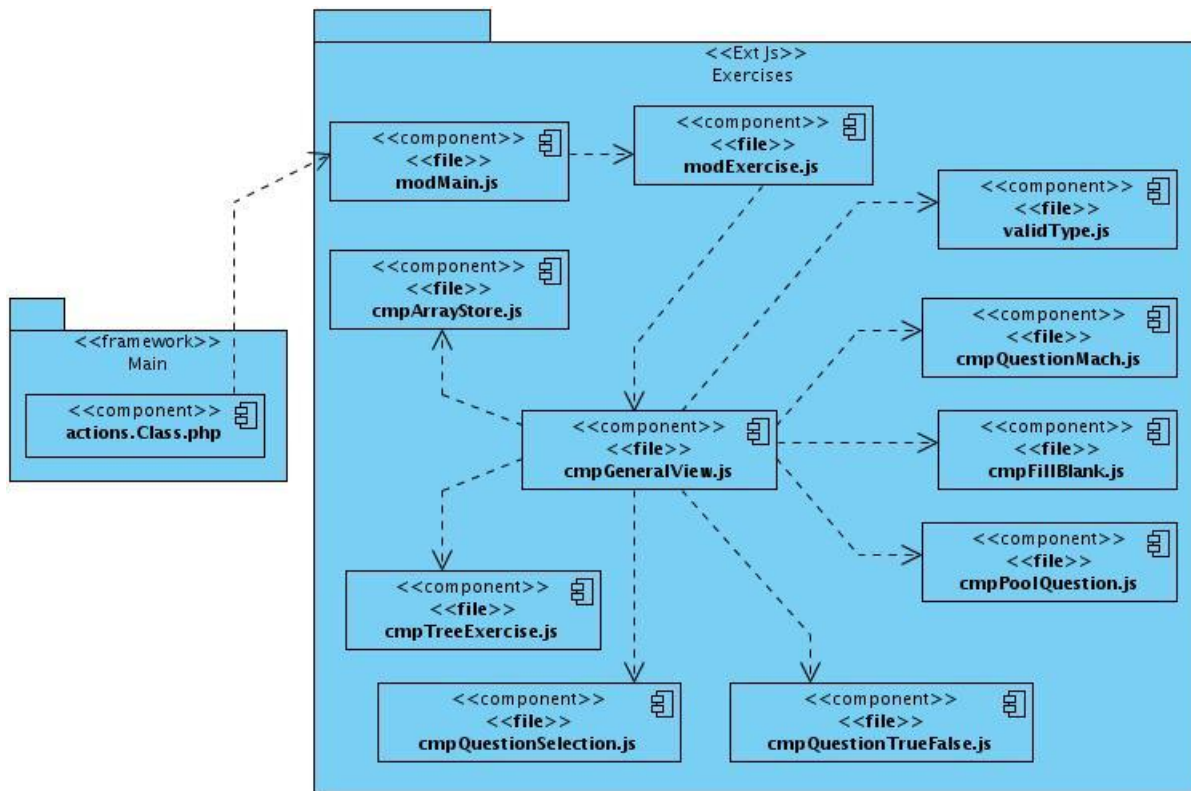


Figura 2.4. Diagrama de componentes general del módulo.

En la figura anterior también se pueden observar el resto de los componentes, los cuales se describirán a continuación para lograr una mejor comprensión del funcionamiento interno del módulo.

modExercise.js: Este componente es el encargado de colapsar los paneles laterales izquierdo y derecho para mostrar la interfaz principal del módulo encabezada por “cmpGeneralView.js”.

cmpGeneralView.js: Constituye la interfaz principal del módulo sirviendo de contenedor de los demás componentes. El mismo se encuentra dividido en 3 secciones. La sección uno contiene los menues y opciones en la parte superior, la sección dos proporciona un espacio para mostrar los ejercicios en el lateral izquierdo y la tercera permite mostrar los demás componentes en el centro.

cmpTreeExercise: Muestra los ejercicios creados por el usuario autenticado en el sistema. Ver figura 2.5.

cmpPoolQuestion.js: Muestra una lista de preguntas con algunos datos relevantes y opciones como eliminar, editar y exportar para cada una de ellas por separado.

Los componentes **cmpQuestionMach**, **cmpQuestionTrueFalse**, **cmpQuestionFillBlank**, **cmpQuestionSelection**, contienen un formulario con todos los campos requeridos por las preguntas de enlazar columna, verdadero o falso, llenar espacios en blanco y (Selección Simple y Múltiple) respectivamente. Además cuentan con todas las funciones requeridas que permiten validar, guardar y editar los datos de forma correcta.

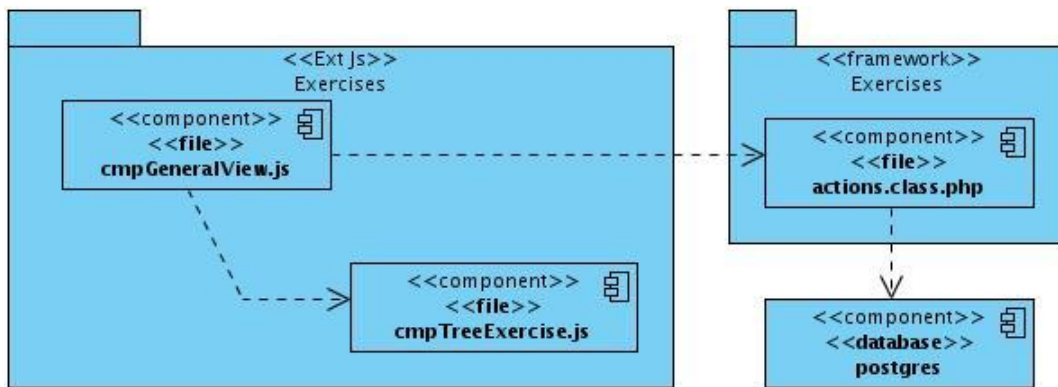


Figura 2.5. Diagrama de componentes mostrar ejercicios.

En el diagrama anterior se presenta la forma en que se obtienen los ejercicios mediante una solicitud por Ajax a la función “getExercise” dentro del action.class.php el cual es el encargado de satisfacer la petición demandada obteniendo los datos de la base de datos.

2.3.2.1 Funcionalidades principales

Guardar, editar y eliminar ejercicio.

Estas funcionalidades se pueden realizar desde el componente principal “cmpGeneralView.js” mediante un menú que puede ser desplegado al dar clic derecho sobre cualquier ejercicio. Ver figura 2.5. Posteriormente se ejecutan las funciones pertinentes para cada una de las acciones en cuestión, que se encuentran en el fichero “actions.class.php”.

Crear nueva pregunta

Partiendo del script “cmpGeneralView.js” se muestran varias opciones que permiten crear una nueva pregunta de cualquier tipo contemplado y asignarla a algún ejercicio previamente creado. Ver figura 2.6.

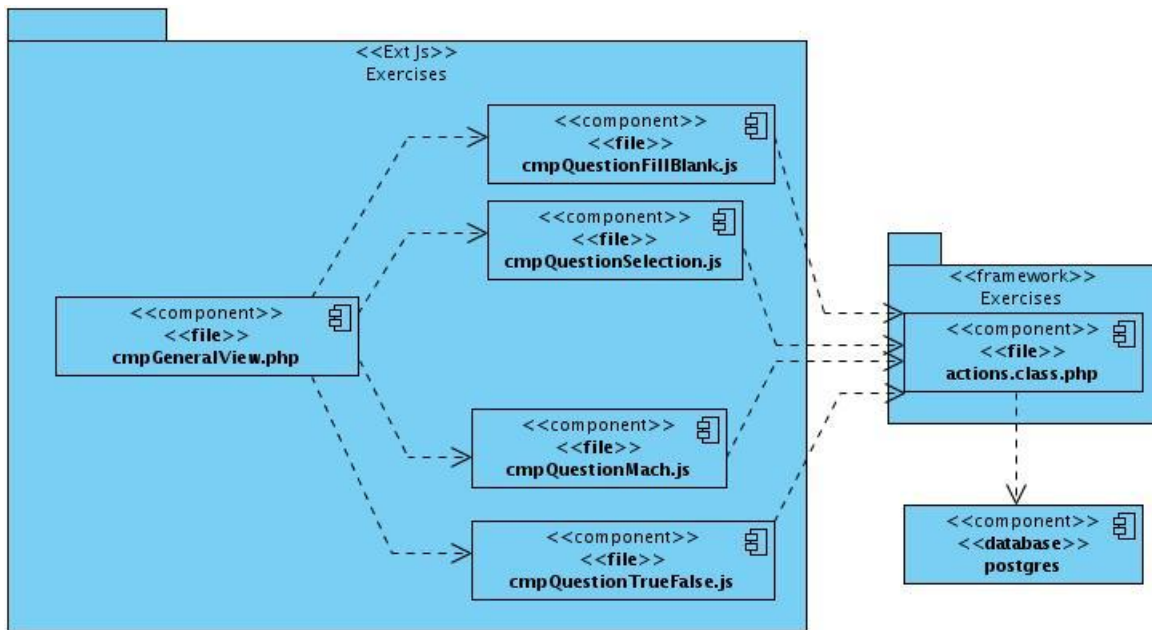


Figura 2.6. Diagrama de componentes crear nueva pregunta.

Editar o eliminar pregunta

Partiendo del componente “cmpGeneralView.js” se pueden visualizar todas las preguntas creadas, ya sean de un ejercicio en específico o de todos, mediante el componente “cmpPoolQuestion”. Este componente visualiza ciertos datos de interés de cada pregunta en forma de tabla, permitiendo al usuario realizar cualquiera de las acciones en cuestión a una pregunta en específico. Ver figura 2.7.

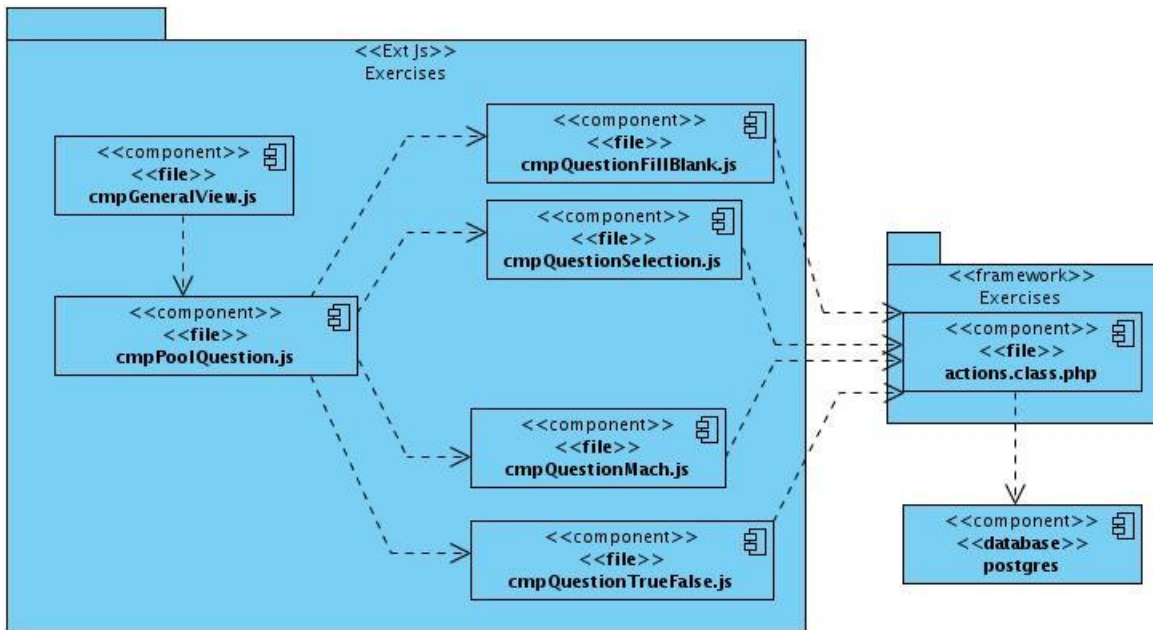


Figura 2.7. Diagrama de componentes gestionar pregunta.

Exportar ejercicios y preguntas

En los dos casos se parte del componente principal, con la diferencia que para exportar el ejercicio la acción se realiza desde este mismo componente y para la pregunta desde el componente "cmpPoolQuestion.js". Ver figura 2.8 y 2.9. A partir de ahí el procedimiento de exportado es el mismo, con la diferencia que para los ejercicios la clase "actions.class.php" devolverá un arreglo de preguntas, las cuales serán exportadas utilizando la especificación QTI en su versión 2.0.

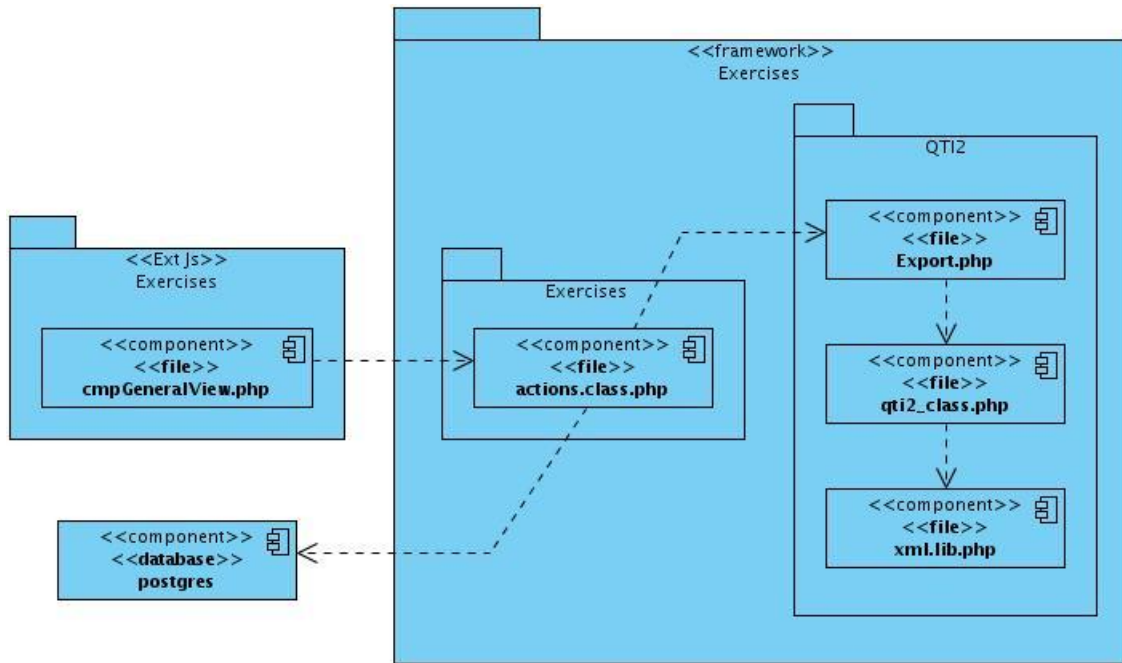


Figura 2.8. Diagrama de componentes exportar ejercicio.

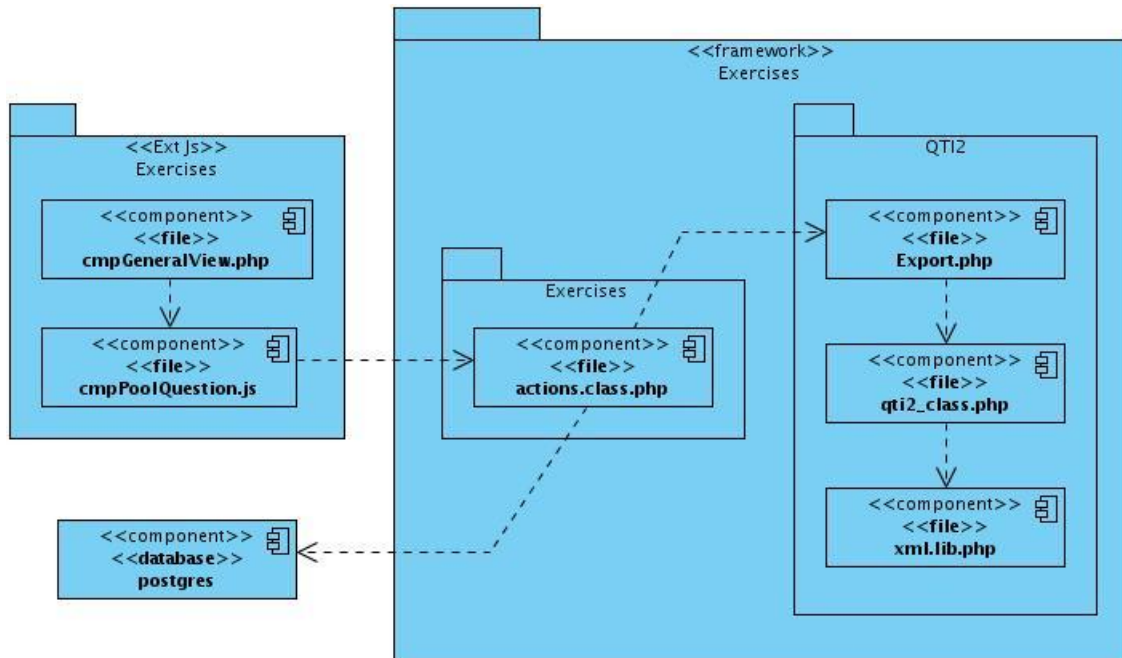


Figura 2.9. Diagrama de componentes exportar pregunta.

2.4 Modelo de despliegue

El modelo de despliegue muestra una representación de los nodos del sistema, en los cuales se llevará a cabo el despliegue de los componentes. El propósito principal de este diagrama es capturar la configuración de los elementos de procesamiento, y las conexiones entre estos elementos en el sistema. Está compuesto por uno o más nodos (elementos de procesamiento con al menos un procesador, memoria, y posiblemente otros dispositivos), dispositivos (nodos estereotipados con una capacidad de procesamiento en el nivel modelado de abstracción), y conectores, entre nodos, y entre nodos y dispositivos.

A continuación se muestra el diagrama de despliegue representando los principales componentes externos requeridos por la aplicación. Ver figura 2.10.

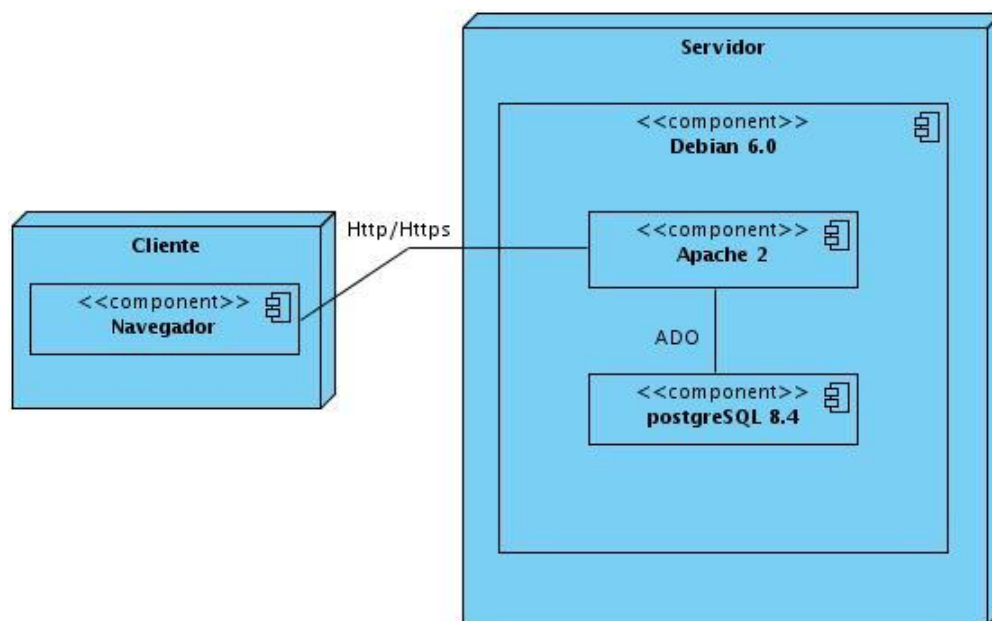


Figura 2.10. Diagrama de despliegue.

Se recomienda que el sistema sea instalado en un servidor apache usando PHP 5 sobre el sistema operativo Debian versión 6.0 y que se use un gestor de bases de datos PostgreSQL 8.4.

2.5 Modelo de datos

CRODA usa básicamente una base de datos relacional al igual que el módulo de ejercicios, el cual se presenta a continuación.

2.5.1 Modelo de base de datos relacional

El esquema relacional del módulo ejercicios cuenta con 8 tablas de las cuales la tabla de “tb_exercise” y de “tb_question” están relacionadas con la tabla “sf_guard_user”. Esta relación permite registrar el usuario que crea la pregunta o el ejercicio en cuestión.

A continuación se describen las tablas con que cuenta este módulo:

1. tb_exercise: para almacenar los datos de los ejercicios que tiene un usuario.
2. tb_question: para guardar los datos generales que tienen las preguntas.
3. r_exercise_question: relaciona los ejercicios con las preguntas.
4. nom_type_question_exercise: almacena los tipos de preguntas soportados.
5. tb_fill_blank: para guardar los datos específicos de las preguntas de llenar espacios en blanco.
6. tb_true_false: para guardar los datos específicos de las preguntas de verdadero o falso.
7. tb_multi_choice: para guardar los datos específicos de las preguntas de respuestas múltiples y selección simple.
8. tb_match_q: para guardar los datos específicos de las preguntas de enlazar columna.

En la figura 2.11 se muestra el diagrama entidad-relación del modelo de datos del módulo.

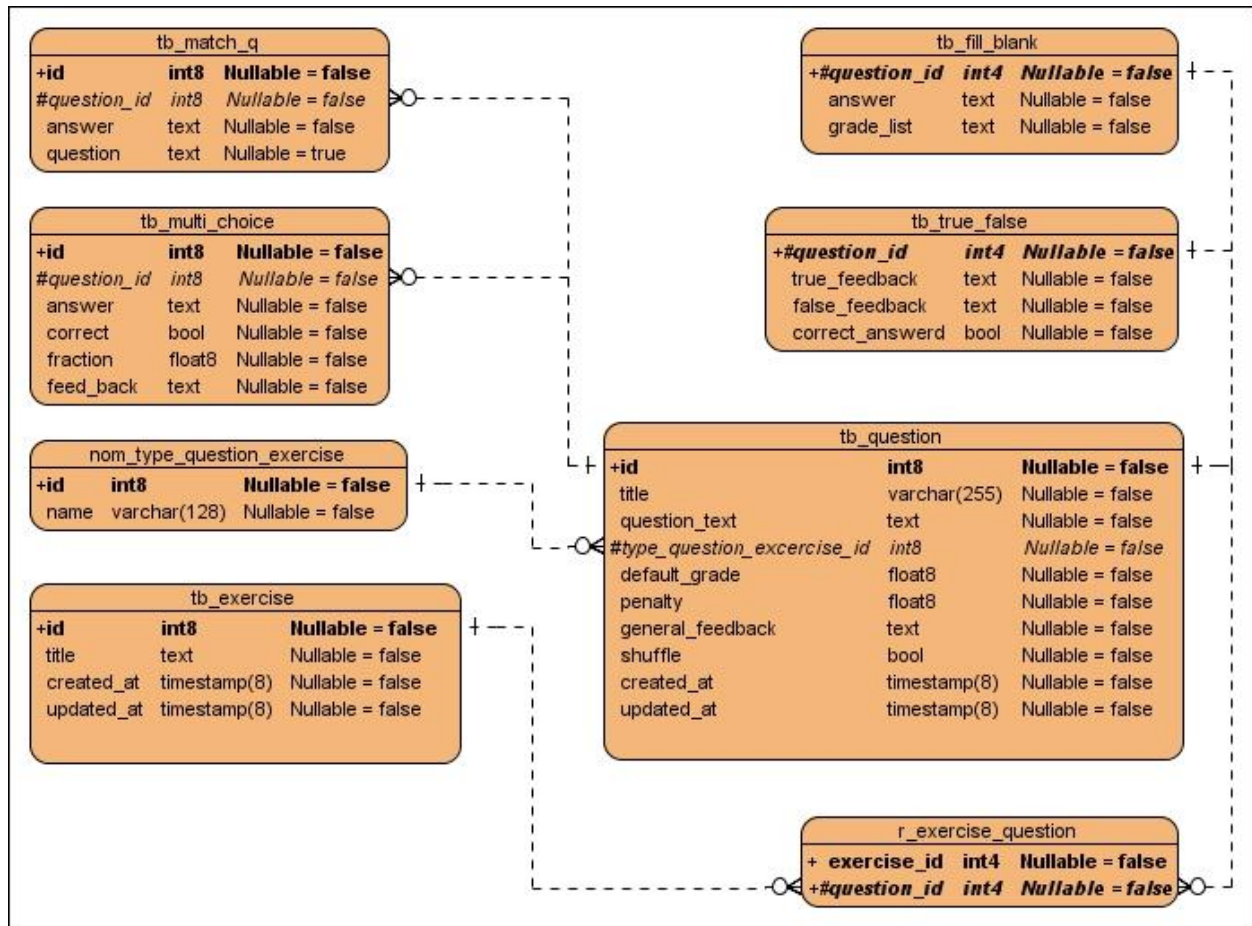


Figura 2.11. Diagrama entidad-relación del módulo ejercicios.

2.6 Estándar de codificación

En pos de garantizar un código más legible, reutilizable y que permita una adecuada comprensión para los otros programadores del proyecto, es que se hace necesario establecer este estándar de codificación.

Para el desarrollo de este módulo se definieron un conjunto de pautas a seguir en cuanto a la forma de escribir código fuente teniendo en cuenta los siguientes aspectos:

Idioma.

Para nombrar variables, funciones, clases y demás se usarán nombres descriptivos y en idioma inglés.

Nomenclatura de los identificadores.

Los identificadores se comienzan a escribir con minúscula y en el caso de componerse de más de una palabra se escriben una a continuación de la otra, comenzando con mayúscula a partir de la segunda. Como se puede apreciar en el fragmento de código que se presenta a continuación en la figura 2.12.

```
1 | class assessmentItem {  
2 |     private $identifier;  
3 |     private $title;  
4 |     private $timeDependent;  
5 | }
```

Figura 2.12. Nomenclatura de los identificadores.

Normas de los comentarios.

Se definió el uso de comentarios claros y precisos que ayudarán a la mejor comprensión del código implementado con el objetivo de aumentar su mantenibilidad a lo largo del tiempo.

En las clases

Para implementar una clase se escribe una pequeña descripción donde se expliquen de forma general los propósitos de la misma, así como su autor y sistema al que pertenece. Dicha descripción se escribe de la siguiente forma:

```
1 | /**  
2 |  * Nombre de la clase *  
3 |  * Descripción *  
4 |  * @author *  
5 |  * @package *(módulo)  
6 |  * @copyright *  
7 |  * @version (versión - parche)  
8 |  */
```

Figura 2.13. Normas de los comentarios en las clases.

En las funciones

Antes de la declaración de funciones críticas o de relevante importancia se establece una pequeña descripción donde se explica el propósito de la misma, su autor, los parámetros que maneja y los valores de retorno. Dicha declaración quedaría como sigue a continuación:

```
1  /**
2  * Nombre de la función *
3  * Descripción *
4  * @author * (en el caso que no sea el mismo autor de la clase)
5  * @param *(los parámetros que recibe la función con su descripción)
6  * @return *(representa los valores de retorno)
7  */
```

Figura 2.14. Normas de los comentarios en las funciones.

Identación.

La indentación es usada para tener una mejor visibilidad en el diseño del código, por lo tanto se tomará en cuenta a la hora de escribir cualquier fragmento de código, utilizándose siempre espacios en blanco y nunca tabulaciones, las estructuras anidadas usarán cuatro espacios para la indentación, además los lexemas que se usan para comenzar y cerrar bloques “{” y “}” no poseerán indentación alguna. Ver figura 2.15.

```
1  public function getQuestion()
2  {
3      $id=$this->getQuestionParameter('id');
4      if ($id == 1)
5      {
6          $this->getParameter();
7      }
8  }
```

Figura 2.15. Identación del código.

2.7 Conclusión

En este capítulo se detallaron los artefactos fundamentales del flujo de implementación. Además se explicaron las principales funcionalidades del módulo, mostrando las vistas correspondientes, así como los diagramas de paquetes y componentes que ilustraron los diferentes scripts y clases principales. También se definieron estándares de codificación utilizados para la implementación del sistema.

CAPÍTULO 3: Validación de la solución propuesta.

3.1 Introducción

Las pruebas tienen una gran importancia en el desarrollo de software puesto que se consideran indispensables para obtener buenos resultados. Se realizan con el objetivo de revisar que todo el software tenga el nivel de calidad requerido. Permiten detectar errores durante el funcionamiento del software, probando así la entrada y salida correcta de datos. Este proceso debe comenzar en la fase de requerimientos y terminar con la finalización de la aplicación con los casos de prueba al sistema ejecutable en la fase de construcción según la metodología escogida para el diseño de la aplicación. En el proceso de pruebas se definen varios métodos, técnicas y tipos de pruebas, las cuales se abordarán a continuación.

3.2 Técnicas de pruebas

Técnica de Caja Blanca: Basada en los requerimientos, en el diseño y en el código fuente.

Entrada: Estándar de programación, archivos fuentes y documentos de diseño.

Salidas: Documentos con defectos encontrados. Estadísticas acerca del seguimiento al estándar, así como de algunas características del sistema, tales como: el tamaño (puntos de función), la complejidad (ciclomática, estructural), la modularidad (cohesión, acoplamiento) y la legibilidad.

Técnica de Caja Negra: Es basada en los requerimientos y verifica si el sistema realmente hace lo que debe hacer, es decir, se llevan a cabo sobre la interfaz del software. Es completamente indiferente el comportamiento interno y la estructura del programa.

Entrada: Archivos ejecutables, documentos de requerimientos.

Salida: Documentos con defectos encontrados. Métricas de la aplicación de casos de pruebas.

Para realizar la validación de la solución propuesta se escogió la técnica de caja negra, para así comprobar si el sistema realmente hace lo que debe hacer.

3.3 Casos de prueba

El propósito principal de un caso de prueba es especificar una forma de probar el sistema, incluyendo las entradas con las que se ha de probar, los resultados esperados y las condiciones bajo las que ha de probarse.

Los requerimientos son la fuente principal para obtener los casos de pruebas pero no son el único medio. Normalmente, un caso de prueba se deriva de un caso de uso en el modelo de casos de uso o de una realización de un caso de uso en el modelo del diseño. Con estos casos de prueba se validan los requerimientos funcionales del sistema. Los siguientes son casos de pruebas comunes:

- Un caso de prueba especifica cómo probar un caso de uso o un escenario específico de un caso de uso. Un caso de prueba de este tipo incluye la verificación del resultado de la interacción entre los actores y el sistema, donde se satisfacen las precondiciones y pos condiciones especificadas por el caso de uso y que se sigue la secuencia de acciones especificadas por el caso de uso.
- Un caso de prueba basado en un caso de uso especifica típicamente una prueba del sistema como “caja negra”, es decir, una prueba del comportamiento observable externamente del sistema.

Se debería desarrollar un caso de prueba para cada escenario del caso de uso. Los escenarios de un caso de uso se identifican describiendo los distintos caminos del flujo básico y flujo alternativo del caso de uso. (Valón., 2010)

3.3.1 Descripción de los casos de prueba.

Gestionar ejercicio

Descripción general:

El CU inicia cuando el generador decide crear, modificar, eliminar o previsualizar un ejercicio. Finaliza cuando la acción sobre el ejercicio es realizada.

Condiciones de ejecución:

El usuario debe estar autenticado en la aplicación y debe poseer el rol de generador.

CAPÍTULO 3 Validación de la solución propuesta

Para editar un Ejercicio, debe estar seleccionado previamente.

Para eliminar un Ejercicio, debe estar seleccionado previamente.

Para previsualizar un Ejercicio, debe estar seleccionado previamente.

Para exportar un Ejercicio, debe estar seleccionado previamente.

| Nombre de la sección | Escenarios de la sección | Descripción de la funcionalidad |
|-------------------------------|---|--|
| SC 1: Adicionar ejercicio. | EC 1.1: Insertar título del ejercicio. | Se inserta el ejercicio con el título introducido. |
| | EC1.2: El campo se deja vacío. | Se muestra un mensaje diciendo que el campo no puede ser nulo o vacío. |
| EC 2: Editar ejercicio. | EC 2.1: Modificar título del ejercicio. | Se modifica el título del ejercicio. |
| | EC 2.2: Se deja el campo vacío. | El título del ejercicio no puede ser nulo. |
| EC 3: Eliminar ejercicio. | EC 3.1: Eliminar ejercicio. | Esta seguro que desea eliminar el ejercicio "nombre". |
| | EC 3.2. Se selecciona OK. | Se elimina el ejercicio deseado. |
| EC 4: Exportar ejercicio. | EC 4.1: Exportar ejercicio. | Se exporta el ejercicio seleccionado. |
| | EC 4.2 Exportar ejercicio que no tiene preguntas asignadas. | El ejercicio no se puede exportar porque no tiene preguntas asignadas. |
| EC 5: Vista previa ejercicio. | EC 5.1 mostrar la vista previa del ejercicio | Se muestra la vista previa del ejercicio seleccionado. |

Tabla 1. Escenarios para el caso de uso "Gestionar ejercicio".

Caso de prueba nuevo ejercicio.

| Escenario | Descripción | Título | Respuesta del sistema | Flujo central |
|-----------|--|--------|--|---------------------------|
| EC 1 | Se selecciona la opción Nuevo ejercicio. | | Desactiva la interfaz principal. Brinda la posibilidad de introducir los datos siguientes: <ul style="list-style-type: none"> • Título del ejercicio | Ejercicio/Nuevo Ejercicio |

CAPÍTULO 3 Validación de la solución propuesta

| | | | | |
|--------|------------------------------------|---|--|------------------------------------|
| | | | y permite: <ul style="list-style-type: none"> • Insertar un ejercicio. • Cancelar la operación en cualquier momento. | |
| EC 1.1 | Introduce el título del ejercicio. | V | Valida los datos. Guarda el ejercicio. El ejercicio se inserta en la parte lateral izquierda. Se activa la interfaz principal. | Ejercicio/Nuevo Ejercicio/OK |
| EC 1.2 | | I | Se muestra un mensaje diciendo que el título del ejercicio no puede ser nulo. | Ejercicio/Nuevo Ejercicio/OK |
| EC 1.3 | Selecciona la opción cancelar. | | Se cierra la interfaz para insertar el título. Se activa la interfaz principal. | Ejercicio/Nuevo Ejercicio/Cancelar |

Tabla 2. Caso de prueba “Nuevo ejercicio”.

Caso de prueba previsualizar ejercicio.

| Escenario | Descripción | Respuesta del sistema | Flujo central |
|-----------|---------------------------------------|--|----------------------------------|
| EC 1 | Selecciona la opción de Vista Previa. | Se abre un nuevo tab con la vista previa del ejercicio seleccionado. | Ejercicio/Ejercicio/Vista Previa |

Tabla 3. Caso de prueba “Previsualizar ejercicio”.

Caso de prueba exportar ejercicio.

| Escenario | Descripción | Respuesta del sistema | Flujo central |
|-----------|---|---|------------------------------|
| EC 1 | Exportar un ejercicio | Muestra una interfaz para proveer la dirección donde se desea guardar el ejercicio a exportar. | Ejercicio/Ejercicio/Exportar |
| EC 1.1 | Exportar un ejercicio que no contiene preguntas asignadas | Se muestra un mensaje diciendo que el ejercicio no puede ser exportado por no contar con preguntas asignadas. | Ejercicio/Ejercicio/Exportar |

Tabla 4. Caso de prueba “Exportar ejercicio”.

Caso de prueba Eliminar ejercicio.

| Escenario | Descripción | Respuesta del sistema | Flujo central |
|-----------|----------------------------|---|---------------------------------|
| EC 1 | Eliminar un Ejercicio | Muestra un mensaje de advertencia para confirmar la eliminación de ejercicio. | Ejercicio/Ejercicio/Eliminar |
| EC 1.1 | Se selecciona la opción OK | El ejercicio es eliminado. | Ejercicio/Ejercicio/Eliminar/OK |

CAPÍTULO 3 Validación de la solución propuesta

Tabla 5. Caso de prueba “Eliminar ejercicio”.

Caso de prueba Editar ejercicio.

| Escenario | Descripción | Título | Respuesta del sistema | Flujo central |
|-----------|--------------------------|--------|--|---|
| EC 1 | Editar ejercicio | | Se activa el campo para editar el título del ejercicio. | Clic sobre el ejercicio/Otro clic sobre el ejercicio. |
| EC 1.1 | Insertar el nuevo título | V | Se modifica el título del ejercicio. | Cualquier acción fuera del campo |
| EC 1.2 | Se deja vacío el campo | I | Se muestra un mensaje diciendo que el campo no puede ser nulo o vacío. | Cualquier acción fuera del campo |

Tabla 6. Caso de prueba “Editar ejercicio”.

Gestionar pregunta de verdadero o falso

A continuación se presenta la tabla de clases de equivalencia tomando en cuenta un dominio de valores, que respondan a un funcionamiento correcto del caso de prueba y otro conjunto que se encuentre fuera de ese rango, con la intención de que devuelva el error especificado.

| Condición externa | Clases de equivalencia válidas. | Clases de equivalencia no válidas. |
|--|---------------------------------|---|
| El campo es obligatorio. | Cualquier valor. | Dejar el campo vacío, es decir el valor nulo. |
| El campo debe especificar un valor numérico. | $0 < \text{valor}$ | $\text{valor} \leq 0$ |

Tabla 7. Partición de equivalencia.

Descripción general:

El CU inicia cuando el generador decide crear, modificar, eliminar o previsualizar una pregunta de verdadero o falso. Finaliza cuando la acción sobre la pregunta es realizada.

Condiciones de ejecución:

El usuario debe estar autenticado en la aplicación y debe poseer el rol de generador.

Para insertar una pregunta, debe estar seleccionado previamente el ejercicio al cual se va a insertar.

CAPÍTULO 3 Validación de la solución propuesta

| Nombre de la sección | Escenarios de la sección | Descripción de la funcionalidad |
|---|---|--|
| EC 1: Nueva Pregunta de verdadero o falso. | EC 1.1: Nueva Pregunta de verdadero o falso. | Se añade la nueva pregunta de verdadero o falso. |
| | EC 1.2: Existen campos inválidos. | El formulario no es válido. |
| | EC 1.3: Existen campos vacíos. | El formulario no es válido. |
| EC 2: Editar Pregunta de verdadero o falso. | EC 2.1: Editar pregunta de verdadero o falso. | Se edita la pregunta. |
| | EC 2.2: Existen campos inválidos. | El formulario no es válido. |
| | EC 2.3: Existen campos vacíos. | El formulario no es válido. |
| EC 3: Eliminar Pregunta de verdadero o falso. | EC 3.1: Eliminar pregunta. | Está seguro que desea eliminar la pregunta. |
| | EC 3.2: Se selecciona ok. | Se elimina la pregunta. |
| EC 4: Exportar Pregunta de verdadero o falso. | EC 4.1: Exportar pregunta. | Se exporta la pregunta. |
| EC 5: Vista previa de la pregunta. | EC 5.1: Clic sobre el título de la pregunta. | Se muestra la vista previa de la pregunta. |

Tabla 8. Escenarios para el caso de uso “Gestionar preguntas de verdadero o falso”.

Caso de prueba Nueva pregunta de verdadero o falso.

| Escenario | Descripción | Nombre | Texto | Nota | Feedback G | Respuesta c | Feedback C | Feedback I | Respuesta del sistema | Flujo central |
|-----------|-------------|--------|-------|------|------------|-------------|------------|------------|-----------------------|---------------|
| | | | | | | | | | | |
| | | | | | | | | | | |

CAPÍTULO 3 Validación de la solución propuesta

| | | | | | | | | | | |
|--------|--------------------------------------|---|---|---|---|---|---|---|--|---|
| EC 1 | Nueva pregunta de verdadero o falso. | | | | | | | | Brinda la posibilidad de introducir los datos siguientes: - Nombre de la pregunta. - Texto de la pregunta. - Nota - Feedback General - Respuesta Correcta. - Feedback Respuesta Correcta. - Feedback Respuesta Incorrecta. y permite: - Insertar una pregunta de Verdadero o Falso. -Cancelar la operación en cualquier momento. | Ejercicio/Nueva Pregunta/Verdadero o Falso |
| EC 1.1 | Insertar datos de la pregunta. | V | V | V | V | V | V | V | Valida los datos. Guarda la pregunta. Muestra un mensaje diciendo que la pregunta se ha guardado correctamente. | Ejercicio/Nueva Pregunta/Verdadero o Falso/Guardar |
| EC 1.2 | Selecciona la opción de cancelar. | | | | | | | | Se cierra el formulario y retorna a la vista principal. | Ejercicio/Nueva Pregunta/Verdadero o Falso/Cancelar |
| EC1.3 | Existen datos incompletos. | I | V | V | V | V | V | V | Muestra un mensaje de advertencia diciendo que el formulario no es válido. Resalta los campos con errores. | Ejercicio/Nueva Pregunta/Verdadero o Falso/Guardar |
| | | V | I | V | V | V | V | V | | |
| | | V | V | I | V | V | V | V | | |
| | | V | V | V | I | V | V | V | | |
| | | V | V | V | V | I | V | V | | |
| | | V | V | V | V | V | I | V | | |
| | | V | V | V | V | V | V | I | | |
| | | I | I | I | I | I | I | I | | |

Tabla 9. Caso de prueba “Nueva pregunta de verdadero o falso”.

Caso de prueba editar pregunta de verdadero o falso.

| Escenario | Descripción | Nombre | Texto | Nota | Feedback G | Respuesta c | Feedback C | Feedback I | Respuesta del sistema | Flujo central |
|-----------|--------------------------------|--------|-------|------|------------|-------------|------------|------------|--|---------------|
| EC 1. | Editar pregunta de verdadero o | | | | | | | | Brinda la posibilidad de editar los datos previamente guardados: | Editar |

CAPÍTULO 3 Validación de la solución propuesta

| | | | | | | | | | | | |
|--------|-------------------------------------|---|---|---|---|---|---|---|---|--|----------|
| | falso. | | | | | | | | | <ul style="list-style-type: none"> - Nombre de la pregunta. - Texto de la pregunta. - Nota - Feedback General - Respuesta Correcta. - Feedback Respuesta Correcta. - Feedback Respuesta Incorrecta. y permite: <ul style="list-style-type: none"> - Editar una pregunta de Verdadero o Falso. - Cancelar la operación en cualquier momento. | |
| EC 1.1 | Modificar los datos de la pregunta. | V | V | V | V | V | V | V | V | Valida los datos. Modifica la pregunta. Muestra un mensaje diciendo que la pregunta se ha editado correctamente. | Guardar |
| EC 1.2 | Selecciona la opción de cancelar. | | | | | | | | | Se cierra el formulario y retorna a la vista principal. | Cancelar |
| EC1.3 | Existen datos incompletos | I | V | V | V | V | V | V | V | Muestra un mensaje de advertencia diciendo que el formulario no es válido. | Guardar |
| | | V | I | V | V | V | V | V | V | | |
| | | V | V | I | V | V | V | V | V | | |
| | | V | V | V | I | V | V | V | V | | |
| | | V | V | V | V | V | I | V | V | | |
| | | V | V | V | V | V | V | I | V | | |
| | | V | V | V | V | V | V | I | V | | |
| | | I | I | I | I | I | I | I | I | | |

Tabla 10. Caso de prueba "Editar pregunta de verdadero o falso".

Caso de prueba eliminar una pregunta.

| Escenario | Descripción | Respuesta del sistema | Flujo central |
|-----------|-----------------------------|---|---------------|
| EC 1 | Eliminar una pregunta. | Muestra un mensaje de advertencia para confirmar la eliminación de la pregunta. | Eliminar |
| EC 1.1 | Se selecciona la opción OK. | La pregunta es eliminada. | Eliminar/OK |

Tabla 11. Caso de prueba "Eliminar pregunta".

Caso de prueba exportar una pregunta.

CAPÍTULO 3 Validación de la solución propuesta

| Escenario | Descripción | Respuesta del sistema | Flujo central |
|-----------|----------------------------------|--|------------------|
| EC 1 | Exportar una pregunta. | Muestra la interfaz para seleccionar la dirección local donde se guardará la pregunta. | Exportar |
| EC 1.1 | Se selecciona la opción Aceptar. | La pregunta es exportada. | Exportar/Aceptar |

Tabla 12. Caso de prueba “Exportar pregunta”.

Caso de prueba previsualizar una pregunta.

| Escenario | Descripción | Respuesta del sistema | Flujo central |
|-----------|-----------------------------|---|-------------------------------------|
| EC 1 | Previsualizar una pregunta. | Se muestra la vista previa de la pregunta | Clic sobre el título de la pregunta |

Tabla 13. Caso de prueba “Previsualizar pregunta”.

3.4 Clasificación de las no conformidades

La descripción de las clasificaciones de las No Conformidades ayudará a la acelerada gestión de los proyectos y a que el proceso de pruebas de liberación se realice en un tiempo reducido y planificado, sin atentar contra el cronograma de desarrollo y los compromisos con el cliente. Se podrá realizar una evaluación más profunda del software a revisar, se tendrán mejores estadísticas de cuáles son las No Conformidades más comunes en los diferentes tipos de software y se le podrá dar un mejor tratamiento a la No Conformidad en el proceso de liberación del software. (Mares Amézquita, 2007)

Una no conformidad según lo planteado en la Norma UNE-EN ISO 9001:2000. “Es el incumplimiento de algunos de los requisitos”. (Mares Amézquita, 2007)

Una No Conformidad es un fallo en el Sistema de gestión de la Calidad que puede producirse por varias razones, no alcanzar el nivel de aceptación establecido en un determinado indicador y errores en la documentación del Sistema. Se trata de una desviación entre lo que hay escrito (lo que hemos dicho que vamos a hacer) y lo que ha ocurrido (lo que hemos hecho). Este fallo queda registrado en un informe y se establecen las acciones preventivas y correctivas necesarias para arreglar lo que no funcione y evitar que vuelva a ocurrir. (Fernández Pereda)

Las mismas se clasifican de acuerdo al nivel de importancia en:

1. **Las No Conformidades Significativas:** Son aquellas que afectan la calidad del producto o servicio de manera visible, impidiendo o no el cumplimiento de algún requisito.
2. **Las No Conformidades No Significativas:** Son aquellas que resultan menos visibles, que no atentan el cumplimiento de algún requisito.
3. **Las Recomendaciones:** Son aquellas que quedan en función de la apreciación del probador para oportunidades de mejoras del producto o servicio.

Las No Conformidades se clasifican además en función de sus características de acuerdo al tipo de artefacto como:

1. **No Conformidades de Documentación:**
 - Formato.
 - Error técnico.
 - Otros errores.
 - Correspondencia con otra documentación.
2. **No Conformidades de Aplicación:**
 - Validación.
 - Opciones que no funcionan.
 - Errores de interfaz.
 - Funcionalidad.
 - Excepciones.
 - Correspondencia de lo implementado con lo documentado.
3. **No Conformidades comunes (para ambos artefactos):**
 - Ortografía.
 - Redacción.
 - Errores de idioma.

3.5 Resultados de las pruebas de caja negra.

Los resultados obtenidos durante las iteraciones fueron los descritos en la tabla 14. En la misma se pueden apreciar la cantidad de no conformidades, las que fueron solucionadas (cerradas) y las que por algún motivo justificado no pudieron ser resueltas.

| Iteración | No conformidades | Cerradas | No proceden |
|-----------|------------------|----------|-------------|
| 1ra | 15 | 15 | 0 |
| 2da | 6 | 7 | 0 |
| 3ra | 2 | 2 | 0 |
| 4ta | 0 | - | - |

Tabla 14. Resultado de las pruebas funcionales.

3.6 Conclusiones

En este capítulo se abordaron los métodos y técnicas existentes para la realización de pruebas. Se explicaron de modo general las pruebas de Caja Blanca y las pruebas de Caja Negra, aplicándose esta última para la validación de la solución propuesta. También se plasmaron los resultados arrojados por dicha técnica, así como algunos de sus artefactos generados para ilustrar la forma en que se llevó a cabo el proceso de prueba.

CONCLUSIONES

Luego de terminar el presente trabajo se arribaron a las siguientes conclusiones:

- El estudio del arte permitió conocer las herramientas que dan soporte a la especificación IMS QTI, las cuales sirvieron de guía para lograr la implementación del módulo.
- El estudio de la especificación IMS QTI, permitió implementar la versión 2.0, creándose los cinco tipos de preguntas de verdadero o falso, selección simple, respuestas múltiples, espacios en blanco y enlazar columna.
- Las pruebas realizadas permitieron validar la solución propuesta arrojando resultados positivos.

Por todo lo anteriormente planteado se concluye que los objetivos propuestos para el presente trabajo han sido cumplidos satisfactoriamente. Se incluyen una serie de recomendaciones que deben tenerse en cuenta para los futuros trabajos.

RECOMENDACIONES

Durante el desarrollo de este trabajo han ido surgiendo nuevas ideas que podrían ser implementadas en el futuro, para lograr un módulo más útil, efectivo y funcional, por lo que se recomienda:

- Implementar otras interacciones dentro de las simples y de texto descritas en el presente trabajo que permitan crear más tipos de preguntas.
- Implementar la especificación IMS Content Packaging para empaquetar las preguntas exportadas y añadirle metadatos.
- Implementar un motor de IMS QTI para lograr una completa visualización y prueba de las preguntas creadas en la herramienta.

REFERENCIAS BIBLIOGRÁFICAS

1. **Annis Parra Rodríguez, Rafael Lorente Salgueiro. 2009.** *Desarrollo de una herramienta de Autor Web para el diseño flexible de objetos de aprendizaje.* La Habana : s.n., 2009.
2. **AulaDiez español online.** AulaDiez español online. *AulaDiez español online.* [En línea] [Citado el: 16 de Enero de 2011.] <http://www.auladiez.com>.
3. **Baltasar Fernández Manjón, Pablo Moreno Ger, José Luis Sierra Rodríguez, Iván. 2006.** *Uso de estándares aplicados a Tic.* 2006.
4. **Benjamín Díaz Tito, Jesús Hidalgo Guillén, Yordanis Gutiérrez Gómez. 2008.** *Herramientas para la Reutilización de Contenidos a partir de la plataforma Moodle.* La Habana : s.n., 2008.
5. **2005.** Campus virtual EHU-UPV. *Campus virtual EHU-UPV.* [En línea] 2005. [Citado el: 20 de Enero de 2011.] <http://campusvirtual.ehu.es/acciones/A2.htm>.
6. *E-learning en Cuba: un ejemplo y un reto para la epidemiología cu.* **Méndez, Dr. José B. Jardines. 2005.** 1, Ciudad de La Habana : Editorial Ciencias Médicas, 2005, Vol. 43.
7. *eXelearning o cómo crear recursos educativos digitales con sencillez.* **Francisco José Navarro García, Beatriz Climent Piqueras. 2009.** 2009, @tic. revista d'innovació educativa, pág. 4.
8. **Fernández Pereda, Héctor.** Norma ISO 9001:2000 Sistema de Gestión de la Calidad y Requisitos. [En línea] [Citado el: 10 de mayo de 2010.] http://www.buscarportal.com/articulos/iso_9001_2000_gestion_calidad.html.
9. **García, Juan Egea. 2005.** Universidad De Murcia. [En línea] Noviembre de 2005. [Citado el: 20 de Enero de 2011.] <http://www.um.es/atiga/gat/tdm/reload/sesion1.pdf>.
10. **Guzmán, Clara López. 2005.** *Los Repositorios de Objetos de Aprendizaje como soporte a un entorno e-learning.* Salamanca : s.n., 2005.
11. **Hector del Castillo Ventosa, Gonzalo García Fernández, Rafael Sanz Sierra. 2004.** *Implementación del módulo.* Madrid : s.n., 2004.
12. **IMS Global Learning Consortium. 2011.** IMS Global Learning Consortium. *IMS Global Learning Consortium.* [En línea] 2011. [Citado el: 23 de Enero de 2011.] <http://www.imsglobal.org/question/>.
13. —. **2006.** *IMS Question and Test Interoperability Assessment Test, Section, and Item Information Model V 2.1.* 2006.

14. —. **2005.** *IMS Question and Test Interoperability Conformance Guide V 2.0.* 2005.
15. —. **2005.** *IMS Question and Test Interoperability Implementation Guide V 2.0.* 2005.
16. —. **2005.** *IMS Question and Test Interoperability Information Model V 2.0.* 2005.
17. —. **2005.** *IMS Question and Test Interoperability Integration Guide V 2.0.* 2005.
18. —. **2005.** *IMS Question and Test Interoperability Meta-data and Usage Data V 2.0.* 2005.
19. —. **2005.** *IMS Question and Test Interoperability Migration Guide V 2.0.* 2005.
20. —. **2005.** *IMS Question and Test Interoperability Overview V 2.0.* 2005.
21. —. **2005.** *IMS Question and Test Interoperability XML Binding V 2.0.* 2005.
22. **Judith Carrodegua Argote, Henry Ricardo Labañino. 2008.** *Análisis y Diseño de una herramienta de autor Web interoperable.* La Habana : s.n., 2008.
23. **Lourdes Costero Ranz, Susana Oliva Pérez, Miguel Ángel Sánchez Fernández. 2006.** *Editor y motor de QTI.* Madrid : s.n., 2006.
24. **Mares Amézquita, Saúl. 2007.** Gestión de No Conformidades. [En línea] 2007. [Citado el: 10 de mayo de 2011.] <http://www.mex.ops-oms.org/contenido/tuberculosis/cdtaller/presentaciones/M%C3%B3dulo%20%20>
25. **Oswaldo Ernesto Stable Vilches, Yandris Mata Cabrera. 2010.** *“Desarrollo de un módulo para la validación de modelos estadísticos en el Paquete de Ayuda a la Toma de Decisiones y Soluciones Integrales (PATDSI).* La Habana : s.n., 2010.
26. **Peñalvo, Francisco José García. 2005.** *Estado actual de los sistemas e-learning.* 2005.
27. **Potencier, François Zaninotto y Fabien.** *Symfony 1.0, la guía definitiva.* s.l. : Apress.
28. Reload reusable eLearning Object Authoring & Delivery. *Reload reusable eLearning Object Authoring & Delivery.* [En línea] [Citado el: 20 de enero de 2011.] <http://www.reload.ac.uk/>.
29. **Respondus.** Testing, Assessment, and Survey Tools for the Leading Course Management Systems. *Testing, Assessment, and Survey Tools for the Leading Course Management Systems.* [En línea] [Citado el: 16 de Marzo de 2011.] <http://www.respondus.com/products/respondus.shtml>.
30. **Shea Frederick, Colin Ramsay, Steve 'Cutter' Blades. 2008.** *Learning Ext JS.* s.l. : Akshara Aware, 2008.

31. *Unidades de aprendizaje, una propuesta de complemento a los objetos de aprendizaje.* **Hernández, Eduardo. 2009.** 2009, Teoría de la Educación "Educación y Cultura en la Sociedad de la Información."
32. **Valón., Eneida García. 2010.** *Herramienta para generar las estadísticas cuantitativas de las No Conformidades.* La Habana : s.n., 2010.
33. **Wiley, David A. 2000.** The Instructional Use of Learning Objects: Online Version. *The Instructional Use of Learning Objects: Online Version.* [En línea] 2000. [Citado el: 17 de enero de 2011.] <http://reusability.org/read/>.

Anexo 1. XML de los tipos de preguntas contemplados.

En este trabajo solo se tomaron en cuenta cinco tipos de preguntas a desarrollar bajo la especificación IMS QTI versión 2.0, de las cuales se presenta a continuación su estructura XML.

Pregunta verdadero/falso

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- This example is from CRODA 2.0 -->
3 <assessmentItem xmlns="http://www.imsglobal.org/xsd/imsqti_v2p0"
4   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5   xsi:schemaLocation="http://www.imsglobal.org/xsd/imsqti_v2p0_imsqti_v2p0.xsd"
6   identifier="choice" title="Pregunta de verdadero o falso" adaptive="false" timeDependent="false">
7   <responseDeclaration identifier="RESPONSE" cardinality="single" baseType="identifier">
8     <correctResponse>
9       <value>True</value>
10    </correctResponse>
11  </responseDeclaration>
12  <outcomeDeclaration identifier="SCORE" cardinality="single" baseType="integer">
13    <defaultValue>
14      <value>0</value>
15    </defaultValue>
16  </outcomeDeclaration>
17  <itemBody>
18    <choiceInteraction responseIdentifier="RESPONSE" shuffle="false" maxChoices="1">
19      <prompt>La palma real es un símbolo patrio, reponda si la afirmación anterior es corecta.</prompt>
20      <simpleChoice identifier="True">verdadero.</simpleChoice>
21      <simpleChoice identifier="False">Falso.</simpleChoice>
22    </choiceInteraction>
23  </itemBody>
24  <responseProcessing
25    template="http://www.imsglobal.org/question/qti_v2p0/rptemplates/match_correct"/>
26 </assessmentItem>

```

Figura A1. XML preguntas de verdadero o falso.

Pregunta selección simple

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- This example is from CRODA 2.0 -->
3 <assessmentItem xmlns="http://www.imsglobal.org/xsd/imsqti_v2p0"
4   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5   xsi:schemaLocation="http://www.imsglobal.org/xsd/imsqti_v2p0_imsqti_v2p0.xsd"
6   identifier="choice" title="selección simple" adaptive="false" timeDependent="false">
7   <responseDeclaration identifier="RESPONSE" cardinality="single" baseType="identifier">
8     <correctResponse>
9       <value>opcionA</value>
10    </correctResponse>
11  </responseDeclaration>
12  <outcomeDeclaration identifier="SCORE" cardinality="single" baseType="integer">
13    <defaultValue>
14      <value>0</value>
15    </defaultValue>
16  </outcomeDeclaration>
17  <itemBody>
18    <choiceInteraction responseIdentifier="RESPONSE" shuffle="false" maxChoices="1">
19      <prompt>De las siguientes afirmaciones seleccione la correcta.</prompt>
20      <simpleChoice identifier="opcionA">José Martí es el Héroe Nacional de Cuba.</simpleChoice>
21      <simpleChoice identifier="opcionB">Antonio Maceo es el Héroe Nacional de Cuba.</simpleChoice>
22      <simpleChoice identifier="opcionC">José Martí no es el Héroe Nacional de Cuba.</simpleChoice>
23    </choiceInteraction>
24  </itemBody>
25  <responseProcessing
26    template="http://www.imsglobal.org/question/qti_v2p0/rptemplates/match_correct"/>
27 </assessmentItem>

```

Figura A2. XML preguntas de selección simple.

Pregunta respuestas múltiples

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- This example is from CRODA 2.0 -->
3 <assessmentItem
4   xmlns="http://www.imsglobal.org/xsd/imsqti_v2p0"
5   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
6   xsi:schemaLocation="http://www.imsglobal.org/xsd/imsqti_v2p0_imsqti_v2p0.xsd"
7   identifier="choiceMultiple"
8   title="Selección múltiple"
9   adaptive="false"
10  timeDependent="false">
11  <responseDeclaration identifier="RESPONSE" cardinality="multiple" baseType="identifier">
12    <correctResponse>
13      <value>H</value>
14      <value>O</value>
15    </correctResponse>
16    <mapping lowerBound="0.0" upperBound="2.0" defaultValue="-2.0">
17      <mapEntry mapkey="H" mappedvalue="1.0"/>
18      <mapEntry mapkey="O" mappedvalue="1.0"/>
19      <mapEntry mapkey="A" mappedvalue="-0.25"/>
20      <mapEntry mapkey="C" mappedvalue="-0.25"/>
21      <mapEntry mapkey="N" mappedvalue="-0.25"/>
22      <mapEntry mapkey="Ca" mappedvalue="-0.25"/>
23    </mapping>
24  </responseDeclaration>
25  <outcomeDeclaration identifier="SCORE" cardinality="single" baseType="integer"/>
26  <itemBody>
27    <choiceInteraction responseIdentifier="RESPONSE" shuffle="true" maxChoices="0">
28      <prompt>
29        seleccione por cuáles de los siguientes elementos está compuesta el Agua.
30      </prompt>
31      <simpleChoice identifier="H">Hidrógeno</simpleChoice>
32      <simpleChoice identifier="A">Azufre</simpleChoice>
33      <simpleChoice identifier="C">Carbono</simpleChoice>
34      <simpleChoice identifier="O">Oxígeno</simpleChoice>
35      <simpleChoice identifier="N">Nitrógeno</simpleChoice>
36      <simpleChoice identifier="Ca">Calcio</simpleChoice>
37    </choiceInteraction>
38  </itemBody>
39  <responseProcessing template="http://www.imsglobal.org/question/qti_v2p0/rptemplates/map_response"/>
40 </assessmentItem>

```

Figura A3. XML preguntas de respuestas múltiples.

Pregunta espacios en blanco

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- This example is from CRODA 2.0 -->
3 <assessmentItem xmlns="http://www.imslobal.org/xsd/imsqti_v2p0"
4   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5   xsi:schemaLocation="http://www.imslobal.org/xsd/imsqti_v2p0 imsqti_v2p0.xsd"
6   identifier="textEntry" title="Pregunta de espacios en blanco" adaptive="false" timeDependent="false">
7   <responseDeclaration identifier="RESPONSE" cardinality="single" baseType="string">
8     <correctResponse>
9       <value>esperanza</value>
10    </correctResponse>
11    <mapping defaultValue="0">
12      <mapEntry mapKey="esperanza" mappedValue="5"/>
13    </mapping>
14  </responseDeclaration>
15  <outcomeDeclaration identifier="SCORE" cardinality="single" baseType="float"/>
16  <itemBody>
17    <p>Identifique la palabra perdida en la frase de José Martí.</p>
18    <blockquote>
19      <p>José Martí dijo:<br/> Los niños son la
20      <textEntryInteraction responseIdentifier="RESPONSE" expectedLength="9"/>
21    del mundo.
22    </p>
23  </blockquote>
24  </itemBody>
25  <responseProcessing template="http://www.imslobal.org/question/qti_v2p0/rptemplates/map_response"/>
26 </assessmentItem>

```

Figura A4. XML preguntas de espacios en blanco.

Pregunta enlazar columna

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- This example is from CRODA 2.0 -->
3 <assessmentItem xmlns="http://www.imslobal.org/xsd/imsqti_v2p0"
4   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5   xsi:schemaLocation="http://www.imslobal.org/xsd/imsqti_v2p0 imsqti_v2p0.xsd"
6   identifier="match" title="Enlazar columna" adaptive="false" timeDependent="false">
7   <responseDeclaration identifier="RESPONSE" cardinality="multiple" baseType="directedPair">
8     <correctResponse>
9       <value>P A</value>
10      <value>V C</value>
11      <value>CE PL</value>
12    </correctResponse>
13    <mapping defaultValue="0">
14      <mapEntry mapKey="P A" mappedValue="1"/>
15      <mapEntry mapKey="V C" mappedValue="1"/>
16      <mapEntry mapKey="CE PL" mappedValue="1"/>
17    </mapping>
18  </responseDeclaration>
19  <outcomeDeclaration identifier="SCORE" cardinality="single" baseType="float"/>
20  <itemBody>
21    <matchInteraction responseIdentifier="RESPONSE" shuffle="true" maxAssociations="4">
22      <prompt>Enlace la columna según corresponda:</prompt>
23      <simpleMatchSet>
24        <simpleAsociableChoice identifier="P" matchMax="1">perro</simpleAsociableChoice>
25        <simpleAsociableChoice identifier="V" matchMax="1">verde</simpleAsociableChoice>
26        <simpleAsociableChoice identifier="M" matchMax="1">mango</simpleAsociableChoice>
27        <simpleAsociableChoice identifier="CE" matchMax="1">cedro</simpleAsociableChoice>
28      </simpleMatchSet>
29      <simpleMatchSet>
30        <simpleAsociableChoice identifier="C" matchMax="4">colores</simpleAsociableChoice>
31        <simpleAsociableChoice identifier="PL" matchMax="4">plantas</simpleAsociableChoice>
32        <simpleAsociableChoice identifier="A" matchMax="4">animales</simpleAsociableChoice>
33      </simpleMatchSet>
34    </matchInteraction>
35  </itemBody>
36  <responseProcessing template="http://www.imslobal.org/question/qti_v2p0/rptemplates/map_response"/>
37 </assessmentItem>

```

Figura A5. XML preguntas de enlazar columna.

GLOSARIO DE TÉRMINOS

Assessment: Es el equivalente a una 'prueba'. Contiene una recopilación de artículos que se utilizarán para determinar el nivel o dominio, que un participante tiene sobre una determinada materia.

Item: Constituye el elemento más pequeño dentro de la especificación IMS QTI. Este no es más que una pregunta con sus instrucciones e interacciones (interactions).

Interaction: Permite la interacción con los items o preguntas, construyéndose de esta manera la respuesta de la pregunta.

SCORM (Sharable Content Object Reference Model): Modelo de referencia para el desarrollo e integración de contenidos educativos.

XML: Un metalenguaje extensible de etiquetas desarrollado.

IMS: Es un consorcio internacional que ha propuesto un conjunto de especificaciones sobre distintos aspectos que intervienen en el modelado del aprendizaje en *e-learning*.

IMS QTI: Contempla una estructura básica que describe la forma de representar preguntas individuales o ítems (assessment item) y gestionar evaluaciones o exámenes completos (assessment).

HTTP (HyperText Transfer Protocol): El protocolo de transferencia de hipertexto usado en cada transacción de la Web.

HTTPs (Protocolo seguro de transferencia de hipertexto): es un protocolo de red basado en el protocolo HTTP, destinado a la transferencia segura de datos de hipertexto.

SQL: Lenguaje de consulta estructurado.

Apache: Servidor HTTP de páginas Web desarrollado por la Apache Software Foundation.

Aplicación: Es el programa que el usuario activa para trabajar en el ordenador. Existen muchos programas de ordenador que pueden clasificarse como aplicación. Generalmente se les conoce como Software.

Browser: Navegador. Herramienta para "Visitar" o visualizar el contenido de las páginas Web y sitios FTP en Internet.

Especificación: Es un documento técnico que describe los componentes (parte estática) y el comportamiento (parte dinámica) de un determinado sistema.

Estructura cliente / servidor: Consiste en que los clientes piden que una tarea sea realizada y el servidor realiza dicha tarea y regresa la información al cliente a través de la red.