



**Universidad de las Ciencias Informáticas
Facultad 4**

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

***Implementación y prueba de la versión 1.1
del módulo Ejercicios de la colección de software
educativo El Navegante en su versión multiplataforma***

Autor: Agustín Castillo Cordero

**Tutores: Ing. Héctor Luis Reyes Pupo
Ing. Sergio Díaz Catalá**

**La Habana, junio de 2012
“Año 54 de la Revolución”**

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor del trabajo “Implementación y prueba de la versión 1.1 del módulo Ejercicios de la colección de software educativo El Navegante en su versión multiplataforma” y autorizo a la Facultad 4 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Autor:

Agustín Castillo Cordero

Tutores:

Ing. Héctor Luis Reyes Pupo

Ing. Sergio Díaz Catalá

RESUMEN

La introducción de las Tecnologías de la Información y las Comunicaciones en el sector de la educación en Cuba han provocado transformaciones en los métodos tradicionales de enseñanza, ya que se han desarrollado por el Ministerio de Educación en Cuba (MINED) varias colecciones de software educativo, entre ellas El Navegante, la cual está dirigida a la enseñanza secundaria.

El desarrollo de una nueva versión multiplataforma de la colección El Navegante para desplegarla en las escuelas venezolanas se lleva a cabo por la Universidad de las Ciencias Informáticas en conjunto con el MINED. Uno de los módulos con los que cuentan los productos de esta colección es Ejercicios, del cual se obtuvo una primera versión con un alto nivel de desarrollo, sin embargo, no se logró satisfacer completamente los requisitos del cliente. En aras de lograr implementar el módulo Ejercicios cumpliendo con las necesidades reales se realiza el presente trabajo. Para dar cumplimiento al objetivo se documentaron los resultados del estudio realizado sobre los sistemas similares existentes y las herramientas y tecnologías utilizadas, se elaboró el Diagrama de Despliegue y los Diagramas de Componentes que conforman el Modelo de Implementación, obteniéndose una descripción de la implementación del sistema. Se realizaron pruebas para validar que el módulo Ejercicios responde a los requisitos planteados, se presenta el diseño de los casos de pruebas y las no conformidades encontradas, las cuales fueron solucionadas. Al concluir, se garantiza la gestión de los cuestionarios interactivos necesarios en el módulo Ejercicios, cumpliendo este con las especificaciones definidas para el mismo.

PALABRAS CLAVE: colección El Navegante, módulo Ejercicios, software educativo

ÍNDICE DE CONTENIDO

INTRODUCCIÓN	1
FUNDAMENTACIÓN TEÓRICA	5
1.1 Software educativo	5
1.2 Análisis de soluciones similares existentes	9
1.3 Tecnologías y herramientas utilizadas	13
1.3.1 Metodología de desarrollo de software	14
1.3.2 Lenguaje de modelado	15
1.3.3 Herramienta para el modelado	16
1.3.4 Sistema gestor de base de datos	16
1.3.5 Lenguajes de desarrollo	17
1.3.6 Framework	20
1.3.7 Entorno de desarrollo integrado	23
1.3.8 Servidor Web	23
1.4 Arquitectura de software	24
1.5 Conclusiones	25
IMPLEMENTACIÓN	26
2.1 Descripción del módulo Ejercicios	26
2.3 Estándar de codificación	27
2.3.1 Indentación y espacios en blanco	27
2.3.2 Convenciones de Nombres (funciones, constantes, variables globales, clases, archivos)	27
2.3.3 Comentarios	29
2.4 Modelo de implementación	30
2.4.1 Diagrama de Despliegue	31

2.4.2 Diagrama de Componentes.....	31
2.5 Conclusiones	39
PRUEBAS	40
3.1 Técnicas de prueba.....	40
3.1.1 Diseño de casos de prueba.....	41
3.2 Resultados de las pruebas.....	64
3.3 Conclusiones	67
CONCLUSIONES GENERALES	68
RECOMENDACIONES.....	69
REFERENCIAS BIBLIOGRÁFICAS.....	70
BIBLIOGRAFÍA.....	73
GLOSARIO DE TÉRMINOS	74

INTRODUCCIÓN

Las Tecnologías de la Información y las Comunicaciones (TIC), se desarrollan a un ritmo acelerado y marcan notablemente la sociedad donde son utilizadas, pues los profundos cambios se manifiestan en todos los ámbitos de la actividad humana, mejorando la calidad de vida de las personas. En Cuba, el uso masivo de las TIC se ha fomentado en todas las esferas como estrategia para beneficiar a la mayor parte posible de la población y de las instituciones, además de facilitar el desarrollo sostenible del país.

Uno de los pilares fundamentales de la Revolución cubana es la educación. La introducción de las TIC en este sector ha provocado un proceso de transformaciones en los métodos tradicionales de enseñanza, desarrollándose amplios programas de elaboración de software educativo, con el fin de elevar la calidad del proceso de enseñanza-aprendizaje. Actualmente las escuelas cubanas disponen de un conjunto de software educativos que apoyan este proceso, siendo el caso de la colección Multisaber para la enseñanza primaria, la colección El Navegante para la enseñanza secundaria y la colección Futuro para la enseñanza preuniversitaria.

Como parte del fortalecimiento de las relaciones de los gobiernos de Cuba y Venezuela, en la X Reunión de la Comisión Intergubernamental Cuba-Venezuela desarrollada en diciembre del 2009, se firmó un contrato con el Ministerio del Poder Popular para la Educación de Venezuela, para la realización del proyecto “Colecciones de Software Educativo Multisaber y El Navegante para Planteles de los Niveles Primaria y Secundaria del Sistema Educativo”, con el objetivo de utilizarlas en las escuelas venezolanas.

Las colecciones Multisaber y El Navegante instaladas en las escuelas cubanas se desarrollaron con herramientas propietarias, tales como Macromedia Director y Toolbook Instructor, por lo que solo pueden ser utilizadas en el sistema operativo Windows. Los software educativos presentes en estas colecciones utilizan aspectos genuinamente cubanos que no son identificados en el entorno venezolano, por lo que se hace necesario contextualizar los contenidos y el diseño, además de migrarlos a software libre, para poder cumplir con las políticas del estado venezolano establecidas en el contrato firmado sobre el uso de tecnologías libres.

A la Universidad de las Ciencias Informáticas (UCI), en conjunto con el Ministerio de Educación (MINED), se le ha asignado la tarea de desarrollar ambas colecciones utilizando tecnologías libres. La colección El Navegante está constituida por 10 productos, en los que se abordan temas de varias asignaturas, tales como Matemática, Geografía e Informática. Cada uno de los productos está estructurado en 6 módulos: Contenidos, Ejercicios, Mediateca, Juegos, Resultados y Maestro(a), tres de los cuales comenzaron a desarrollarse en enero de 2010, obteniéndose una primera versión funcional de ellos.

Uno de los módulos iniciados en la primera etapa de desarrollo fue Ejercicios, el cual alcanzó un alto nivel de avance, sin embargo, no logró cumplir con todos los requisitos, debido a que aún no permite asignar cuestionarios interactivos a un equipo de trabajo integrado por 2, 3 o 4 estudiantes; no permite el seguimiento de la interacción de los estudiantes con el módulo. La herramienta que gestiona los cuestionarios no permite agregar, eliminar o modificar medias (imágenes, videos, sonidos) en los mismos, así como gestionar cuestionarios interactivos de las tipologías enlazar columnas, ordenar textos, seleccionar textos, completar por escritura, seleccionar palabras y dividir en sílabas, además el módulo Ejercicios no permite realizar cuestionarios de las tipologías ordenar textos, seleccionar textos, completar por escritura y seleccionar palabras.

Como resultado de previas investigaciones se obtuvo el análisis y diseño de las funcionalidades que se necesitan incluir en el módulo Ejercicios. Por tanto, es necesario resolver el siguiente problema: ¿Cómo lograr que el módulo Ejercicios de la colección El Navegante en su versión multiplataforma gestione las tipologías de cuestionarios interactivos no desarrolladas en la primera versión, y permita la interacción del usuario con estos?

Del problema a resolver expresado anteriormente, se deriva como **objeto de estudio** el software educativo dedicado a la visualización y gestión de cuestionarios interactivos.

Se plantea como **objetivo general** de la investigación realizar la implementación y prueba de la versión 1.1 del módulo Ejercicios de la colección El Navegante en su versión multiplataforma.

El **campo de acción** está enmarcado en el desarrollo de la versión 1.1 del módulo Ejercicios de la colección El Navegante en su versión multiplataforma.

Para dar cumplimiento al objetivo general se definen los siguientes **objetivos específicos**:

- Realizar un estudio del estado del arte sobre el objeto de estudio de la presente investigación.
- Implementar las funcionalidades a incluir en el módulo Ejercicios de la colección El Navegante en su versión multiplataforma.
- Realizar pruebas a la versión 1.1 del módulo Ejercicios de la colección El Navegante en su versión multiplataforma.

Para dar cumplimiento a los objetivos planteados se definen las siguientes **tareas investigativas**:

- Elaboración de los fundamentos teóricos que sustentan la implementación de la versión 1.1 del módulo Ejercicios.
- Realización del modelo de implementación de la versión 1.1 del módulo Ejercicios.
- Implementación de los componentes de la versión 1.1 del módulo Ejercicios.
- Realización de pruebas al módulo Ejercicios.

Se plantea como **idea a defender**, que si se realiza la implementación y prueba de la versión 1.1 del módulo Ejercicios, se logrará que dicho módulo cumpla con las especificaciones del cliente y las necesidades del usuario final, además la herramienta gestionará los contenidos necesarios para que los cuestionarios interactivos cumplan con las especificaciones definidas para los mismos.

A continuación, se mencionan los **métodos científicos** utilizados:

Teóricos

- **Análítico-Sintético**: Es utilizado para estudiar conceptos relacionados con el software educativo y buscar información referente a los tipos de software educativo existentes, así como para analizar las características de los sistemas similares, las herramientas y tecnologías que se utilizan en el desarrollo del módulo.
- **Análisis Histórico-Lógico**: Es utilizado para analizar la evolución de la colección de El Navegante en su versión multiplataforma, principalmente el desarrollo del módulo Ejercicios, además para realizar un estudio del estado del arte de soluciones similares existentes.

Empíricos

- Observación: Para obtener información sobre las diferentes formas de proceder para solucionar cuestionarios interactivos en el módulo Ejercicios de la colección El Navegante instalada en las escuelas cubanas.

La presente investigación, para un mejor entendimiento, ha sido estructurada en 3 capítulos, los cuales se mencionan a continuación:

Capítulo 1. “Fundamentación teórica”. Se abordan elementos teóricos relacionados con el objeto de estudio. Se realiza un análisis de los sistemas similares existentes en el mundo, además se realiza un estudio de las tecnologías y herramientas utilizadas en el desarrollo del módulo Ejercicios de la colección El Navegante en su versión multiplataforma.

Capítulo 2. “Implementación”. Se describe el funcionamiento del módulo Ejercicios y se modelan el Diagrama de Despliegue y los Diagramas de Componentes que conforman el Modelo de Implementación, obteniéndose una descripción de la implementación del sistema. Además se presentan los estándares de codificación utilizados en la generación de código.

Capítulo 3. “Pruebas”. Se selecciona la técnica y el método de prueba utilizado. Se describen los casos de prueba diseñados con el objetivo de encontrar la mayor cantidad de fallos en el sistema y corregirlos. Además se presenta el resultado que se obtuvo al aplicar las pruebas.

Capítulo 1

FUNDAMENTACIÓN TEÓRICA

El surgimiento del software educativo estuvo propiciado por el desarrollo de la tecnología educativa, un punto de partida fue la aparición de las máquinas de enseñanza en 1920, diseñadas para que los estudiantes ejercitaran y practicaran eligiendo una respuesta entre varias opciones. En la década del 50 se alcanzó un mayor desarrollo en estas máquinas, permitiendo a los estudiantes componer sus respuestas, además las preguntas eran diseñadas en una secuencia programada de lo más fácil a lo más difícil. A partir del funcionamiento de estas máquinas surgió la enseñanza programada, método mediante el cual el estudiante aprendía progresivamente, siendo él el responsable de su aprendizaje, a su propio ritmo y según sus posibilidades, sin la mediación directa de un profesor, a través de máquinas puramente mecánicas y juegos educativos. Luego surgió la enseñanza asistida por computadora, la cual adoptó principios de la enseñanza programada para apoyar el proceso de enseñanza – aprendizaje. Un medio didáctico creado con la finalidad específica de ser utilizado en este proceso es el software educativo. (1)
(2)

1.1 Software educativo

El Dr. Pere Marqués utiliza los términos software educativo, programas didácticos y programas educativos como sinónimos para designar genéricamente los programas para computadoras creados con la finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y de aprendizaje. (3)

Los investigadores Ysauro González Neri, Victoria Carmona Martínez y Sara Espíritu Reyes plantean que “El software educativo es un producto tecnológico diseñado para apoyar procesos educativos, dentro de los cuales se concibe como uno de los medios que utiliza quien enseña y quien aprende para alcanzar determinados propósitos. Además, este software es un medio de presentación y desarrollo de contenidos educativos, como lo puede ser un libro o un video, formato expresivo y secuencia narrativa. De esta manera, el software educativo puede ser visto como un producto y también como un medio.” (4)

Capítulo 1: Fundamentación teórica

El autor considera que un software educativo es un producto creado con el objetivo de apoyar el proceso de enseñanza aprendizaje, que permite al alumno interactuar con los contenidos educativos de una manera entretenida y fácil.

Los programas educativos pueden tratar diferentes materias (matemática, idiomas, geografía...), todos comparten cinco características esenciales: (3)

- Son materiales elaborados con una finalidad didáctica.
- Utilizan la computadora como soporte en el que los alumnos realizan las actividades que ellos proponen.
- Son interactivos, contestan inmediatamente las acciones de los estudiantes y permiten un diálogo y un intercambio de informaciones entre la computadora y los estudiantes.
- Individualizan el trabajo de los estudiantes, ya que se adaptan al ritmo de trabajo de cada uno y pueden adaptar sus actividades según las actuaciones de los alumnos.
- Son fáciles de usar. Los conocimientos informáticos necesarios para utilizar la mayoría de estos programas son mínimos, aunque cada programa tiene unas reglas de funcionamiento que es necesario conocer.

El software educativo al presentar distintas características tiene varias clasificaciones. Según Álvaro Galvis Panqueva en su publicación Ingeniería de Software Educativo, teniendo en cuenta el enfoque educativo y las funciones educativas que asumen se clasifican en:

Algorítmico: Software educativo en el que predomina el aprendizaje vía transmisión de conocimiento, desde quien sabe hacia quien lo desea aprender y el diseño de las actividades de aprendizaje conduce al aprendiz desde donde está hasta donde desea llegar. Se consideran software educativos de tipo algorítmico los siguientes: (5)

- **Sistemas tutoriales:** Incluye las cuatro fases que deben formar parte de todo proceso de enseñanza - aprendizaje: la fase introductoria, en la que se genera la motivación y se centra la

Capítulo 1: Fundamentación teórica

atención en lo que se desea que el alumno aprenda; la fase de orientación inicial, en la que los estudiantes retienen los conocimientos; la fase de aplicación, en la que se recuerda lo aprendido; y la fase de retroalimentación en la que se demuestran y se refuerzan los conocimientos. (5)

- **Sistemas de ejercitación y práctica:** Se parte de la base de que antes de interactuar con el software, el alumno ya adquirió los conceptos y destrezas que va a practicar, permite que los estudiantes resuelvan una gran variedad de ejercicios aplicando lo aprendido y que obtengan más conocimientos acerca de los contenidos ejercitados. (5)

Heurístico: Software educativo en el que predomina el aprendizaje por descubrimiento. El alumno debe llegar al conocimiento a partir de la experiencia, creando sus propios modelos de pensamiento, se pueden comprobar a partir del mismo software los conocimientos adquiridos por el alumno. Se consideran software educativos de tipo heurístico los siguientes: (5)

- **Simuladores y juegos educativos:** Ambos pretenden lograr el aprendizaje por descubrimiento. En una simulación el alumno resuelve problemas, aprende procedimientos, llega a entender las características de los fenómenos y cómo controlarlos, aprende qué acciones tomar en diferentes circunstancias. Los juegos pueden o no simular la realidad, se caracterizan por proveer situaciones entretenidas o retos. Los juegos educativos buscan que dicho entretenimiento sirva de contexto al aprendizaje de algo, dependiendo de la naturaleza del juego. (5)
- **Sistemas expertos:** Son sistemas de computación con capacidades para representar y razonar acerca de algún dominio rico en conocimientos, para resolver problemas y dar consejo a quienes no son expertos en la materia. La capacidad de razonar como un experto es lo que hace a los sistemas expertos particularmente útiles para que los aprendices ganen experiencia en dominios en que es necesario obtenerla. (5)

Sistemas tutoriales inteligentes: No se pueden ubicar en una sola de las dos grandes categorías de software educativo, ya que se caracterizan por mostrar un comportamiento inteligente adaptativo, es decir, adapta el tratamiento educativo en función de aquello que se desea aprender y de las características y

Capítulo 1: Fundamentación teórica

desempeño del alumno. La idea básica es la de ajustar la estrategia de enseñanza - aprendizaje, el contenido y forma de lo que se aprende a los intereses, expectativas y características del alumno. (5)

Según José Antonio Gutiérrez Lam en su artículo *El proyecto de Software Educativo. Algunas consideraciones*, son considerados software educativos los **libros electrónicos**, estos presentan información al estudiante a partir del uso de textos, gráficos, animaciones, videos, con un nivel de interactividad y motivación que le facilite las acciones que realiza. (6)

En Cuba se han creado Centros de estudios de software educativo en todos los Institutos Superiores Pedagógicos y como consecuencia del desarrollo alcanzado el Ministerio de Educación ha desarrollado el concepto de **hiperentorno de enseñanza - aprendizaje** (HEA), el cual consiste en una combinación de diferentes tipos de software educativo, basados en tecnología hipermedia. (7)

La tecnología hipermedia surgió mediante la integración del hipertexto y la multimedia, es la organización de imágenes, videos, textos, sonidos y cualquier otro medio digitalizado en diversos nodos interconectados mediante enlaces, los nodos pueden ser accedidos de forma no lineal. La hipermedia facilita la interacción del usuario con la información de una manera amigable y sencilla. (8)

La colección El Navegante, actualmente en desarrollo, integra varias formas de presentación de la información: textos, imágenes, videos, sonidos y animaciones, las cuales permiten mantener la motivación e interés del alumno en el estudio. La colección tiene como objetivos que los estudiantes aprendan contenidos nuevos de una manera interactiva; contribuir al desarrollo de habilidades mediante la ejercitación y práctica; evaluar los conocimientos de los estudiantes sobre determinados temas o contenidos; promover el aprendizaje de manera entretenida a través de juegos educativos; simular experimentos referentes a temáticas de las asignaturas mediante laboratorios virtuales que apoyen a los estudiantes en la solución de problemas y entender fenómenos semejantes a la realidad. Los productos de dicha colección tienen una estructura de navegación que permite a los estudiantes interactuar con la información según sus necesidades y preferencias, es decir, en el orden que desean. Teniendo en cuenta que en la colección El Navegante están presenten varios tipos de software educativos como: libro

Capítulo 1: Fundamentación teórica

electrónico, sistema de ejercitación y práctica, juegos educativos y simuladores, se puede llegar a la conclusión de que esta colección es un hiperentorno de enseñanza – aprendizaje.

1.2 Análisis de soluciones similares existentes

Actualmente existen aplicaciones que permiten realizar cuestionarios interactivos y gestionan los contenidos de los mismos, los principales aspectos a tener en cuenta son: las tipologías de cuestionarios que gestionan, que sean libres, multiplataforma y que los cuestionarios puedan contar con recursos multimedia, a continuación se caracterizan algunas de estas aplicaciones.

1.2.1 iTest

Es una herramienta de evaluación online que permite a profesores configurar exámenes y a los estudiantes realizarlos utilizando únicamente un navegador Web, cada uno de los exámenes se genera y corrige automáticamente de manera instantánea, desde una base de datos de preguntas creadas y gestionadas por cada profesor. La herramienta muestra la calificación obtenida en pantalla una vez finalizado el examen. (9)

iTest posibilita también la configuración de exámenes que incluyan archivos multimedia como imágenes, sonidos, animaciones y partituras musicales en los enunciados de las preguntas y en las respuestas de manera muy rápida e intuitiva, además se pueden incluir fórmulas y gráficas de funciones matemáticas. Esto hace que iTest sea una herramienta muy versátil y adaptable a las distintas asignaturas y a todos los niveles educativos. (9). Esta herramienta es de Licencia GNU/ Libre, se puede utilizar en los sistemas operativos: Mac OS X, Linux/Unix, Microsoft Windows. (10).

La principal desventaja que presenta esta herramienta es que no permite crear las preguntas de los exámenes por tipología.

1.2.2 QuizFaber

Permite generar diversos tipos de ejercicios interactivos y aplicar en ellos las ventajas de los materiales multimedia, pues se pueden insertar imágenes, sonidos y videos. Permite crear ejercicios de selección

Capítulo 1: Fundamentación teórica

múltiple, respuesta abierta, verdadero/falso, rellenar espacios, unir elementos y personalizar algunos detalles de su diseño, como colores y fuentes. (11)

Es un software gratuito para Windows, compatible con los navegadores actuales, se pueden crear exámenes en varios idiomas. Tiene una versión portable. Se puede ejecutar en discos extraíbles. (12)

1.2.3 Hot potatoes

Es un programa gratuito, puede ser utilizado para cualquier propósito o proyecto que se desee. No es de código abierto, cuenta con diferentes componentes los cuales se describen a continuación: (13)

- **JQuiz:** Permite elaborar ejercicios basados en preguntas. Cada *test* puede estar formado por un número ilimitado de preguntas. Se pueden formular cuatro tipos de preguntas:
 - **Elección múltiple:** Los alumnos han de elegir una respuesta, si es incorrecta el alumno podrá seguir seleccionando opciones hasta dar con la respuesta correcta.
 - **Respuestas cortas:** Los alumnos deberán escribir una respuesta en una casilla de texto.
 - **Pregunta híbrida:** Es una combinación de una pregunta de respuestas múltiples y una pregunta de respuesta corta. Al alumno se le presenta una casilla de texto, si no responde correctamente después de un número de intentos, la pregunta cambia a otra de respuestas múltiples para que sea más.
 - **Multi-selección:** Consiste en que el alumno ha de seleccionar varias respuestas de una lista definida previamente.
- **JCloze:** Se utiliza para elaborar ejercicios de completar espacios en blanco.
- **JMatch:** Se utiliza para crear ejercicios de enlazar. Se disponen una serie de elementos a un lado y cada uno de ellos ha de enlazarse con los elementos dispuestos en el lado opuesto.
- **JCross:** Se utiliza para realizar crucigramas.

1.2.4 Ardora

Es una aplicación informática para docentes, que permite crear actividades para los alumnos de un modo muy sencillo. Se pueden crear más de 45 tipos distintos de actividades, de interés para la investigación se

Capítulo 1: Fundamentación teórica

encuentra completar por escritura, además contiene más de 10 tipos distintos de páginas multimedia: galerías, panorámicas o *zooms* de imágenes, reproductores mp3¹, entre otras. (14)

Ardora es gratuita, siempre y cuando sea usado de forma personal, sin carácter lucrativo y con fines estrictamente educativos. No es un programa de código abierto. (14)

1.2.5 Moodle

Es un Sistema de Gestión de Cursos de Código Abierto (Open Source Course Management System, CMS), conocido también como Sistema de Gestión del Aprendizaje (Learning Management System, LMS) o como Entorno de Aprendizaje Virtual (Virtual Learning Environment, VLE). Es una aplicación web gratuita que los educadores pueden utilizar para crear sitios de aprendizaje efectivo en línea. (15)

Moodle se ofrece libremente como software de código abierto (bajo la Licencia Pública GNU). Es multiplataforma, se puede ejecutar en Windows y sistemas operativos Mac y en las diferentes distribuciones de Linux (por ejemplo Red Hat, Debian o Ubuntu). (15)

Los tipos de preguntas que se pueden presentar en un cuestionario de este sistema son seis: selección simple, selección múltiple (emparejamiento), verdadero o falso, numérico, respuesta corta o rellena los blancos. (16)

1.2.6 El Navegante

La colección El Navegante instalada en las escuelas secundarias de Cuba cuenta con 10 productos, en cada uno están presentes 6 módulos: Temas, Ejercicios, Juegos, Biblioteca, Registro y Maestro. El módulo Ejercicios permite realizar ejercicios de las tipologías: selección simple, selección múltiple, completamiento por desplazamiento y escritura, verdadero y falso, seleccionar textos, seleccionar palabras, pero no brinda la posibilidad de modificar los existentes ni agregar nuevos. Esta colección fue desarrollada con herramientas propietarias y solo puede ser usada en el sistema operativo Windows.

¹ Formato de compresión de archivos audio. Posibilita almacenar música de alta calidad ocupando un mínimo espacio.

Capítulo 1: Fundamentación teórica

1.2.7 Zera

Como parte de la investigación se entrevistaron varios profesores del proyecto Alfaomega², que participan en el desarrollo de la plataforma educativa Zera, acerca de sus características, principales funcionalidades, ventajas y desventajas, específicamente del módulo Ejercicios, realizándose el siguiente resumen:

Zera es una plataforma educativa desarrollada en el Centro de Tecnologías para la Formación (FORTES) de la UCI, permite a estudiantes, profesores, padres/tutores y directivos realizar acciones como consultar una materia o contenido educativo, realizar ejercicios, gestionar contenidos, entre otras. Esta plataforma es capaz de adaptarse a los programas de estudio de los centros educacionales de la enseñanza media superior, siguiendo el concepto de hiperentornos de aprendizaje. Se divide en seis (6) subsistemas, los cuales se encuentran estrechamente interrelacionados, además de dos (2) simuladores desarrollados con el fin de apoyar el aprendizaje de las asignaturas de Física y Matemática. Posee un módulo para la gestión de ejercicios que posibilita la creación de exámenes para evaluar el aprendizaje de los estudiantes. Con el fin de lograr una interoperabilidad de los ejercicios gestionados por la plataforma se utiliza la especificación IMS-QTI³ versión 2.0.

En la actualidad, las prestaciones de la plataforma Zera como producto dirigido a la gestión de ejercicios para el aprendizaje, resultan insuficientes atendiendo a que:

- Escasa interoperabilidad en la fase inicial de la creación del módulo, no se tuvo en cuenta la asociación de sonidos, videos, diaporamas y animaciones como medias de ayuda.
- Solo gestiona las tipologías de ejercicio selección simple, selección múltiple, completar por escritura, completar por desplazamiento, ordenar procedimiento, ordenar medias, enlazar, clasificar texto y clasificar media, interacción gráfica múltiple, interacción gráfica simple.

² El proyecto Alfaomega tiene como propósito general desarrollar una plataforma para la gestión del aprendizaje, la cual fue bautizada con el nombre ZERA.

³ Propone normas para el intercambio de ejercicios entre aplicaciones sin importar su autoría u origen de los mismos.

Capítulo 1: Fundamentación teórica

- La manera de seleccionar las palabras en los ejercicios de completar por desplazamiento y escritura es poco intuitiva y está expuesta a errores.

Las funcionalidades del módulo Ejercicios de la plataforma Zera no están resumidas a un *plugin* o complemento⁴, por lo que no es autónomo, el modelo de datos utilizado en el almacenamiento de la información de los ejercicios es completamente distinto. Este módulo gestiona una variada información que no es de importancia hoy para la solución (lenguaje, palabras clave, autor).

Análisis general de los sistemas similares consultados

Luego de analizadas las características de los sistemas similares consultados se observó que no son capaces de dar solución al problema existente, debido a que no gestionan todas las tipologías de cuestionarios necesarias en el módulo Ejercicios de la colección El Navegante en su versión multiplataforma. En el caso de QuizFaber, Ardora, Hot Potatoes, y El Navegante (versión instalada en las escuelas cubanas) no son de código abierto, por lo que no permiten agregar las tipologías de cuestionarios que no gestionan. iTest no brinda la posibilidad de crear las preguntas de los exámenes por tipología. Zera no permite adicionar, eliminar o modificar medias en los cuestionarios interactivos, además teniendo en cuenta que las funcionalidades del módulo Ejercicios de esta plataforma no están resumidas a un *plugin* y que el modelo de datos utilizado en el almacenamiento de la información de los ejercicios es completamente distinto, se hace muy costoso reutilizar esta solución. En general, las aplicaciones estudiadas no satisfacen las características que debe tener el módulo Ejercicios de la colección de software educativo El Navegante, por lo que se decide desarrollar una versión de este.

1.3 Tecnologías y herramientas utilizadas

Teniendo en cuenta que las tecnologías y herramientas utilizadas en el desarrollo de los productos que conforman la colección El Navegante se encuentran definidas en el documento de arquitectura del proyecto “Arquitectura de la versión multiplataforma de la colección de software educativo El Navegante”, se analiza a continuación las características fundamentales que propiciaron su elección y los beneficios que aportan al módulo Ejercicios.

⁴ Un complemento es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica.

Capítulo 1: Fundamentación teórica

1.3.1 Metodología de desarrollo de software

“Un proceso define quién está haciendo qué, cuándo, y cómo alcanzar un determinado objetivo”. (17)

Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software, guía los esfuerzos de las personas implicadas en el proyecto, explicando los pasos necesarios para terminar el proyecto. (17)

Las metodologías según su filosofía de desarrollo se clasifican en dos grupos: en tradicionales o ágiles, las tradicionales se caracterizan por una fuerte planificación, una documentación exhaustiva y un control del proyecto durante todo el proceso de desarrollo, entre las más utilizadas se encuentran Rational Unified Process (RUP) y Microsoft Solutions Framework (MSF). Las metodologías ágiles están especialmente indicadas en el desarrollo de software donde los requisitos no estén bien definidos o cambien con mucha frecuencia, ya que es adaptable al cambio, están dedicadas a proyectos pequeños, de corta duración, en los que exista una estrecha comunicación entre el cliente y los desarrolladores. Entre estas metodologías se destacan Extreme Programming (XP) y Scrum. (18)

RUP es la metodología que guía el proceso de desarrollo del módulo Ejercicios, debido a que no existe una comunicación directa e inmediata con los clientes, toda la interacción es a través de terceros, esta metodología es adaptable a las necesidades de cada organización, permite obtener una completa documentación del proceso de desarrollo, aspecto muy importante, pues la documentación obtenida en la primera etapa de desarrollo del módulo Ejercicios sirve de apoyo al proceso de desarrollo de la nueva versión, ya que proporciona un mejor entendimiento de sus características y de su funcionamiento. Además, el desarrollo iterativo facilita la reusabilidad de partes implementadas, asegura que todos los requerimientos se implementen correctamente y se corrigen los errores en cada iteración. RUP ofrece una mejor comprensión del sistema debido a la gran cantidad de artefactos que se generan durante el proceso de desarrollo.

El Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación,

Capítulo 1: Fundamentación teórica

diferentes tipos de organizaciones y diferentes niveles de aptitud; utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un software. (17)

Lo que hace único al Proceso Unificado de Desarrollo es que se caracteriza por ser dirigido por caso de uso, a partir de los mismos se detallan los requisitos funcionales del sistema y se construyen los artefactos resultantes del análisis, diseño, implementación y prueba de un software; es centrado en la arquitectura, la cual es una vista del diseño completo con las características más importantes resaltadas, permite que el equipo de trabajo se enfoque en los aspectos más importantes para su construcción, además RUP se distingue por ser iterativo e incremental ya que propone realizar un software en iteraciones, en las que se identifican, especifican, diseñan e implementan los casos de uso más relevantes. (17)

1.3.2 Lenguaje de modelado

Un lenguaje proporciona un vocabulario y unas reglas para permitir la comunicación. Un lenguaje de modelado se centra en la representación gráfica de un sistema, capturando las partes esenciales del mismo. El Lenguaje Unificado de Modelado (UML) sirve para visualizar gráficamente sistemas complejos, tanto en el diseño del software como de la arquitectura hardware donde se ejecutan, siendo esta la razón fundamental por la que se decide utilizar UML como lenguaje para la representación gráfica y documentación de los artefactos generados en el desarrollo del módulo. (19)

Los objetivos de UML son muchos, pero se pueden sintetizar sus funciones: (19)

- Visualizar: Permite expresar de una forma gráfica un sistema de forma que otro lo puede entender.
- Especificar: Permite especificar cuáles son las características de un sistema antes de su construcción.
- Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: Los propios elementos gráficos sirven como documentación del sistema desarrollado y pueden ser utilizados para su futura revisión.

El lenguaje UML tiene una notación gráfica muy expresiva que permite representar en mayor o menor medida todas las fases de un proyecto informático, desde el análisis con los casos de uso, el diseño con los diagramas de clases, hasta la implementación y configuración con los Diagramas de Despliegue. (19)

Capítulo 1: Fundamentación teórica

UML aporta beneficios al desarrollo del módulo Ejercicios, ya que permite realizar una representación gráfica del mismo con un gran rigor en la especificación, lo que proporciona un mayor entendimiento de sus características y facilita su proceso de desarrollo.

1.3.3 Herramienta para el modelado

Las herramientas CASE (por sus siglas en inglés: Computer Aided Software Engineering, o Ingeniería de Software Asistida por Computadora) son aquellas aplicaciones que brindan ayuda al desarrollo de software, disminuyendo su costo en términos de tiempo y de dinero. Garantizan una correcta documentación, generación de código y reutilización del software.

Visual Paradigm for UML

Es una herramienta CASE profesional que soporta el ciclo de vida completo del desarrollo de software: análisis y diseño orientados a objetos, construcción, pruebas y despliegue. Permite realizar todos los tipos de diagramas de clases, código inverso, generar código desde diagramas y generar documentación, una de sus características más importantes es que es multiplataforma. (20)

Para el modelado de la aplicación se utiliza la herramienta CASE Visual Paradigm en su versión 6.4, debido fundamentalmente a que es multiplataforma y que facilita la construcción de sistemas con calidad y de forma rápida, ofrece funcionalidades como la generación de código para varios lenguajes de programación, dentro de estos PHP, generación automática de bases de datos a partir de un Modelo entidad-relación e interoperabilidad entre diagramas. Además genera documentación del sistema en formato HTML y PDF y exporta los diagramas en formatos de imagen.

1.3.4 Sistema gestor de base de datos

Un sistema gestor de base de datos se define como el conjunto de programas que administran y gestionan la información contenida en una base de datos. Ayuda a realizar las siguientes acciones: (21)

- Definición de los datos.
- Mantenimiento de la integridad de los datos dentro de la base de datos.
- Control de la seguridad y privacidad de los datos.
- Manipulación de los datos.

Capítulo 1: Fundamentación teórica

“Entre los diferentes tipos de gestores de base datos existentes se pueden encontrar Oracle, PostgreSQL, MySQL, MS SQL Server, por solo mencionar algunos. Dado que muchos de estos gestores son propietarios y su costo es elevado en el mercado, una alternativa libre sería el uso de gestores como MySQL y PostgreSQL que en los últimos años han alcanzado un nivel de madurez equivalente a los proporcionados por los sistemas propietarios en cuanto a seguridad y eficiencia” (22).

PostgreSQL

Es un potente sistema de base de datos relacional, de código abierto. Cuenta con más de 15 años de desarrollo activo y una arquitectura probada que se ha ganado una sólida reputación de confiabilidad, integridad de datos y corrección. Tiene soporte completo para claves foráneas, uniones, vistas y procedimientos almacenados. Se incluye la mayoría de tipos de datos SQL: 2008, incluyendo INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, y TIMESTAMP. También soporta almacenamiento de objetos binarios, como imágenes, sonidos y video. Posee una documentación excepcional. (23)

PostgreSQL está publicado bajo la licencia BSD (Berkeley Software Distribution), es dirigido por una comunidad de desarrolladores y organizaciones comerciales, las cuales trabajan en su desarrollo. Es altamente escalable, tanto en la enorme cantidad de datos que puede manejar como en el número de usuarios concurrentes que puede acomodar.

Se utiliza PostgreSQL en su versión 8.4 debido a que se distribuye bajo la licencia BSD, la cual permite su uso sin limitaciones, es más eficiente que otros sistemas de base de datos cuando el tamaño de la base de datos es grande, muy estable y mantiene una impecable integridad de los datos; por lo que es muy confiable. Además es considerado el sistema de bases de datos de código abierto más avanzado del mundo, es compatible con el almacenamiento de objetos binarios, incluyendo imágenes, sonidos y videos.

1.3.5 Lenguajes de desarrollo

Un lenguaje de programación es aquel elemento dentro de la informática que permite crear programas mediante un conjunto de instrucciones, operadores y reglas de sintaxis; que pone a disposición del programador para que este pueda comunicarse con los dispositivos hardware y software existentes. (24)

Capítulo 1: Fundamentación teórica

Los lenguajes de programación usados en el desarrollo de aplicaciones web se clasifican en: (25)

- Lenguajes de programación del lado del cliente: Son los lenguajes que basan su procesamiento en el cliente web, es decir que se ejecutan en el navegador del usuario.
- Lenguajes de programación del lado del servidor: Son los lenguajes que se procesan en el lado del servidor y que generan la página antes de enviarla al cliente. Este proceso consiste en el procesamiento de una petición de un usuario mediante la interpretación de un script en el servidor para generar páginas Web dinámicamente como respuesta.

En el desarrollo del módulo Ejercicios se utilizan los lenguajes del lado del cliente XHTML, CSS y JavaScript y como lenguaje del lado del servidor PHP.

XHTML

Lenguaje de Marcado de Hipertexto Extensible (XHTML) es un conjunto de documentos basados en XML que nace como extensión del HTML 4.0 y que permite la interoperabilidad entre aplicaciones basadas en XML. Se trata de un paso adelante en el campo del desarrollo Web, al aprovecharse las ventajas del XML, mantiene la compatibilidad de las aplicaciones de usuario realizadas con versiones distintas. (26)

Con el uso de este lenguaje se garantiza que el módulo Ejercicios tenga gran compatibilidad entre los navegadores que cumplen con estándares Web⁵, garantizando que en posteriores versiones de los navegadores funcione correctamente. Presenta buena facilidad de actualización ya que al crear un XHTML correcto y darle los estilos desde una hoja de estilo CSS (Cascading Style Sheets) es posible crear aplicaciones Web totalmente diferentes sin modificar absolutamente nada del XHTML. Además, aumenta la velocidad de navegación ya que las páginas Web se construyen con mayor rapidez en los navegadores.

CSS

CSS (por sus siglas en inglés: Cascading Style Sheets, en español Hojas de estilos en Cascadas) es un lenguaje de hojas de estilo creado por el W3C que permite a los desarrolladores controlar los aspectos de

⁵ Conjunto de reglas normalizadas que tienen por finalidad conseguir que las tecnologías que conforman la Web sean interoperables, eficientes, confiables, accesibles y fáciles de usar.

Capítulo 1: Fundamentación teórica

estilo y formato de múltiples páginas (X)HTML simultáneamente, permitiendo separar el contenido de presentación en todo momento. (26)

El empleo de hojas de estilo CSS es una práctica que aporta una serie de beneficios como: (26)

- Aumento de la legibilidad y reducción del peso de las páginas Web.
- Mejora del mantenimiento y actualización de los sitios Web.
- Mejora la accesibilidad ya que se pueden definir hojas de estilo locales en función de las necesidades o preferencias del usuario.

El lenguaje CSS da la posibilidad de tener el control del diseño en la aplicación, ya que como se ha dicho, es muy útil para separar los contenidos del diseño, logrando que los documentos electrónicos definidos con XHTML creados sean más flexibles y fáciles de mantener.

JavaScript

Es un lenguaje de programación que se utiliza principalmente para crear páginas Web dinámicas. Técnicamente, es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos, en otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. (27)

JavaScript es seguro y fiable ya que los scripts tienen capacidades limitadas por razones de seguridad, es ejecutado en el cliente por lo que el servidor no es utilizado más de lo debido, aunque tiene la desventaja que el código del script debe descargarse completamente para poder ejecutarlo.

Este lenguaje brinda la posibilidad tener el control total de las animaciones, los mensajes de aviso al usuario, las acciones que se activan al manipular componentes en las páginas Web de la aplicación, lo que facilita la programación de los cuestionarios interactivos de forma dinámica.

PHP

PHP, cuyas siglas responden a un acrónimo recursivo (PHP: Hypertext Preprocessor), es un lenguaje de código abierto muy popular, especialmente adecuado para desarrollo Web y que puede ser incrustado en

Capítulo 1: Fundamentación teórica

HTML. Con PHP se puede procesar la información de formularios, generar páginas con contenidos dinámicos, o enviar y recibir *cookies*. (28)

Las características que distinguen a este lenguaje y que le hacen uno de los más potentes lenguajes de programación Web son: (28)

- Soporta la mayoría de servidores Web de hoy en día, incluyendo Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros.
- Soporta gran cantidad de bases de datos: Adabas D, dBase, Empress, Hyperwave, IBM DB2, Informix, Ingres, InterBase, FrontBase, mSQL, Direct MS-SQL, MySQL, ODBC, Oracle (OCI7 y OCI8), Ovrimos, PostgreSQL, SQLite, Solid, Sybase, Velocis, Unix dbm. .
- Integración con varias bibliotecas externas, permite generar documentos en PDF (documentos de Acrobat Reader) hasta analizar código XML.
- Soportado por una gran comunidad de desarrolladores, como producto de código abierto,
- PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente.

Una de las ventajas que brinda el lenguaje PHP al utilizarlo en el módulo Ejercicios es la posibilidad de programar orientado a objetos, lo cual proporciona mucha claridad al desarrollo. Es uno de los lenguajes de programación Web que más documentación tiene, posee variedad de funciones ya implementadas y de mucha utilidad para los desarrolladores. De manera general, las principales características que distinguen a PHP son: su rapidez; su facilidad de aprendizaje; su soporte multiplataforma tanto de diversos sistemas operativos, como de servidores HTTP y de bases de datos; y el hecho de que se distribuye de forma gratuita. Es utilizada la versión 5.2 de este lenguaje.

1.3.6 Framework

El término *framework* se refiere a una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, se puede considerar como una aplicación genérica incompleta y configurable a la que se puede añadir las últimas piezas para construir una aplicación concreta. Los objetivos principales que persigue un *framework* son: acelerar el proceso de

Capítulo 1: Fundamentación teórica

desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. (29)

Un *framework* Web, se puede definir como un conjunto de componentes que forman un diseño reutilizable que facilita y agiliza el desarrollo de sistemas Web.

El uso de un *framework*: (30)

- Simplifica el desarrollo de una aplicación mediante la automatización de algunos de los patrones utilizados para resolver las tareas comunes.
- Proporciona estructura al código fuente, forzando el desarrollador a crear códigos más legibles y fáciles de mantener.
- Facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas.

En el desarrollo del módulo Ejercicios se utilizan como *framework* de apoyo a la implementación en el lado del cliente jQuery y en el lado del servidor Symfony.

Symfony

Symfony es un *framework* para construir aplicaciones Web con PHP, posee un enorme conjunto de herramientas y utilidades que simplifican el desarrollo de las aplicaciones Web. Es uno de los *framework* PHP más populares entre los usuarios y las empresas, ya que permite que los programadores sean mucho más productivos a la vez que crean código de más calidad y más fácil de mantener. Symfony es maduro, estable, profesional y es muy rico en documentación. (31)

Algunas de las características más importantes de Symfony son: (30)

- Fácil de instalar y configurar en la mayoría de plataformas.
- Independiente del sistema gestor de bases de datos.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la Web.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.

Capítulo 1: Fundamentación teórica

- Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros.

Es utilizado Symfony en su versión 1.4.3 como *framework* de apoyo a la implementación en el lado del servidor porque optimiza el desarrollo de aplicaciones Web. Sigue las mejores prácticas y patrones de desarrollo para la Web, como la separación estricta de código y diseño; usa el patrón Modelo Vista Controlador (MVC). Es muy seguro ya que incluye por defecto protección contra ataques de tipo Cross Site Scripting (XSS) y Cross Site Request Forgery (CSRF). Provee clases y herramientas enfocadas a reducir el tiempo de desarrollo. Además contiene una interfaz de línea de comando que con solo introducir una secuencia de comando se ejecutan tareas que disminuyen el tiempo de desarrollo.

jQuery

jQuery es una librería de JavaScript muy rápida y ligera que simplifica el desarrollo de la parte del cliente de las aplicaciones Web, incluyendo muchas utilidades para crear fácilmente las páginas Web de las aplicaciones dinámicas complejas. (32)

El uso de jQuery permite a los desarrolladores el ahorro de líneas de código ya que implementa una serie de clases que permiten programar sin tener en cuenta el navegador del cliente. Esta librería facilita y acelera la implementación del módulo Ejercicios, pues brinda ayuda en la creación de interfaces de usuario complejas, en la animación del contenido de los cuestionarios interactivos de forma dinámica, mediante el manejo de eventos, comunicación asíncrona AJAX (acrónimo de Asynchronous JavaScript And XML, JavaScript asíncrono y XML) que mejora la interactividad porque se pueden realizar cambios en las páginas sin necesidad de recargarlas, estos beneficios proporcionan una mejor usabilidad por parte del usuario.

jQuery tiene licencia para uso en cualquier tipo de plataforma, está soportado por una gran comunidad de creadores de *plugins* o componentes, lo que permite encontrar fácilmente soluciones ya creadas. Está bien documentado y es muy estable ya que cuenta con una gran comunidad de desarrolladores encargados de la actualización y mejora del mismo. Es utilizada en el módulo Ejercicios la versión 1.4.

Capítulo 1: Fundamentación teórica

1.3.7 Entorno de desarrollo integrado

Un Entorno de Desarrollo Integrado o en inglés Integrated Development Environment (IDE) es una herramienta que permite a los desarrolladores de software escribir sus programas en uno o más lenguajes. Un IDE consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. Los IDEs pueden ser aplicaciones por si solas o pueden ser parte de aplicaciones existentes. (33)

NetBeans IDE

NetBeans es una herramienta para escribir, compilar, depurar y ejecutar programas. Permite a los desarrolladores crear aplicaciones de escritorio, para la Web y para teléfonos móviles, tiene soporte para varios lenguajes como PHP, JavaScript, HTML, CSS y su versión 6.8 tiene soporte para el *framework* Symfony. Es un producto libre y gratuito sin restricciones de uso, está disponible para los sistemas operativos Windows, Mac, Linux y Solaris. Es apoyado por una gran comunidad de desarrolladores y ofrece una variada selección de *plugins* de terceros. (34)

Se utiliza como IDE NetBeans ya que presenta soporte nativo a los *frameworks* jQuery y Symfony, es compatible con PHP5 por lo que propone un esqueleto para organizar el código fuente, el editor integra los lenguajes XHTML, CSS y JavaScript, además es software libre, multiplataforma, incluye resaltado de sintaxis, autocompletado, ayuda, control de versiones y gestión de ficheros.

1.3.8 Servidor Web

Un servidor Web es un programa que se utiliza para atender y responder a las diferentes peticiones de los usuarios por medio de los navegadores, proporcionando todos los recursos que soliciten y mostrándolos en una página Web; los servidores Web atienden y sirven peticiones de los navegadores usando los protocolos HTTP (HyperText Transfer Protocol) y el protocolo HTTPS (versión cifrada y autenticada). (33). Entre los servidores Web más utilizados a nivel mundial se encuentran Apache, Internet Information Server (IIS) y el Xitami.

Apache

El servidor Web Apache es gratuito, desarrollado por el Apache Server Project (Proyecto Servidor Apache) cuyo objetivo es la creación de un servidor Web fiable, eficiente y fácilmente extensible con código fuente

Capítulo 1: Fundamentación teórica

abierto. Este proyecto es conjuntamente manejado por un grupo de voluntarios localizados alrededor del mundo que a través de internet planean y desarrollan el servidor y la documentación relacionada con este. Estos voluntarios son conocidos como el grupo Apache. (35)

Las principales ventajas que proporciona el uso de este servidor Web en el desarrollo del modulo ejercicio son: (35)

- Su licencia. Esta es de código abierto del tipo (Berkeley System Distribution, (BSD)) que permite utilizar esta herramienta con fines comercial y no comercial.
- Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado *script* cuando ocurra un error en concreto.
- Es un servidor altamente configurable de diseño modular. Actualmente existen muchos módulos para Apache que son adaptables y permiten su instalación facilitando el trabajo con este servidor Web.
- Es robusto y seguro.

Puede ser utilizado en una multitud de sistemas operativos como, GNU/Linux, Microsoft Windows, Macintosh y otras, lo que lo hace prácticamente universal, trabaja con gran cantidad de lenguajes. Todas estas ventajas que brinda lo convierten en una herramienta robusta, estable y muy configurable, por lo cual es utilizado en el desarrollo del módulo Ejercicios.

1.4 Arquitectura de software

El patrón arquitectónico Modelo Vista Controlador (MVC) es muy utilizado en el desarrollo de aplicaciones Web, ya que brinda la posibilidad de organizar el código de una aplicación separando las interfaces de usuario, la lógica de control y los datos en capas diferentes.

Cada capa realiza funciones específicas: (30)

- El modelo representa la información con la que trabaja la aplicación, es decir, los datos.
- La vista transforma el modelo en una página Web que permite al usuario interactuar con ella.
- El controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.

Capítulo 1: Fundamentación teórica

Como se mencionó anteriormente el *framework* Symfony usa el patrón MVC, desarrollar una aplicación siguiendo esta arquitectura ofrece múltiples ventajas, pues define una separación estricta entre datos y presentación, lo que permite crear fácilmente distintas representaciones de los mismos datos y reutilizar componentes. Este patrón facilita el mantenimiento de los sistemas, permite que cada componente se especialice en su función, brinda la facilidad de desarrollar aplicaciones rápidas, sencillas y escalables.

1.5 Conclusiones

En este capítulo se analizaron algunos de los conceptos relacionados con el software educativo, teniendo en cuenta además las características de los sistemas similares que permiten realizar cuestionarios interactivos y gestionan los contenidos de los mismos, para conocer las tendencias actuales de su desarrollo, identificándose que ninguno de los consultados cumple con las características del módulo Ejercicios de la colección El Navegante en su versión multiplataforma.

Se realizó una investigación sobre las tecnologías y herramientas utilizadas en el desarrollo del módulo Ejercicios. RUP es la metodología que guía el proceso de desarrollo, UML el lenguaje de modelado y como herramienta CASE Visual Paradigm. Para el desarrollo, que está basado en una arquitectura Modelo Vista Controlador se utilizan los lenguajes XHTML, CSS, JavaScript y PHP, además de los *frameworks* Symfony y jQuery, empleando el IDE NetBeans, el sistema gestor de base de datos PostgreSQL y el servidor Web Apache.

Capítulo 2

IMPLEMENTACIÓN

En el presente capítulo se describe el funcionamiento del módulo Ejercicios. Se especifica cómo los elementos del diseño se implementan en términos de componentes y la distribución de los mismos en el sistema, mediante el Modelo de Implementación que está constituido por los Diagramas de Componentes y el Diagrama de Despliegue, además se describen los estándares de codificación utilizados en la aplicación.

2.1 Descripción del módulo Ejercicios

El módulo Ejercicios cuenta con dos aplicaciones, la herramienta encargada de gestionar los contenidos educativos de los cuestionarios interactivos (*backend*) y la aplicación dedicada a la visualización de los cuestionarios y la interacción con los mismos (*frontend*), que tiene como principales funcionalidades las siguientes:

- Seleccionar cuestionarios de diferentes contenidos y de tres formas (secuencial, al azar o asignados).
- Asignar cuestionarios a un equipo de trabajo integrado por 2, 3 o 4 estudiantes.
- Realizar cuestionarios interactivos de las tipologías seleccionar palabras, completar por escritura, seleccionar y ordenar textos.
- Revisar la respuesta de un cuestionario.
- Consultar los resultados obtenidos por un usuario específico.
- Registrar la traza de la interacción de los usuarios con el módulo.

Previo a este trabajo se desarrolló una investigación, en la cual se realizó el levantamiento de requisitos, así como el análisis y diseño del sistema, los requisitos pueden ser consultados en el Anexo 1.

2.3 Estándar de codificación

Uno de los aspectos fundamentales que se debe tener en cuenta a la hora de implementar un software, es el uso de estándares de codificación, estos son reglas que se definen con el objetivo de establecer un estilo de programación homogéneo, ayudan a crear un código más legible, reutilizable y de fácil mantenimiento, además permiten una mejor comprensión del código por parte de otros desarrolladores.

Teniendo en cuenta los lenguajes de programación y los *frameworks* utilizados, se detallan a continuación los estándares definidos para la generación de código del módulo Ejercicios.

2.3.1 Indentación y espacios en blanco

Alineación

Se utiliza dos espacios como unidad de alineación en vez de utilizar la tecla TAB (cuatro espacios). No hay senderos con espacios en blanco.

Longitud de la línea

Se evita utilizar líneas con más de 80 caracteres de longitud, estas no son bien manejadas por algunas herramientas y terminales.

Líneas plegadas

Cuando una expresión no cabe en una línea simple debido a su extensión se divide en más de una línea, siguiendo las siguientes precisiones:

- Dividir después de una coma.
- Dividir después de un operador.
- Alinear la nueva línea al inicio de la expresión en el mismo nivel que la línea anterior.

2.3.2 Convenciones de Nombres (funciones, constantes, variables globales, clases, archivos)

Convenio de nombres

Los nombres que se utilizan son cortos y descriptivos, facilitando el entendimiento del código generado. Se utiliza comentarios en todos los casos que sean normados y en caso de que el programador lo considere necesario. Para los nombres se establecen las siguientes reglas:

Funciones y métodos

La notación **Camel** consiste en escribir los identificadores con la primera letra de cada palabra en mayúsculas y el resto en minúscula: **Ej. IdEjercicio**.

Las funciones y métodos se nombran usando el formato camelCase (notación camello) que es una variante de la notación Camel donde la primera letra es minúscula. **Ej. mostrarEjercicio()**

Constantes

El código de Symfony no incluye ninguna constante, ya que utiliza su propio objeto para almacenar la configuración, llamado **sfConfig**, y que reemplaza a las constantes.

Variables globales

Si se necesita definir variables globales, el nombre comenzará con guión bajo simple.

Clases

Los nombres de las clases comienzan con el **sufijo** del módulo, seguido del formato camelCase: Ej. **ejMediador**. Las clases que genera Doctrine comienzan con:

Tb_: las clases de entidad.

nom_: las clases de nomencladores.

r_: las clases que representan relaciones muchos-a-muchos.

Atributos

Los atributos se nombran siguiendo el formato camelCase. **Ej. getData()**

Los miembros privados de las clases (dígase aquellos que se usan solo en la misma clase donde son declarados; PHP aún no soporta espacios de nombres privados realmente obligatorios) comienzan con guión bajo. **Ej. \$_status, _sort(), _initTree()**

Nombre de las tablas en la base de datos

Los prefijos que se utilizan en la creación de tablas son los siguientes:

tb_ En tablas que almacenan la mayor cantidad de características de una entidad.

nom_ En tablas nomencladoras.

r_ Para establecer la relación de mucho a mucho entre tablas (cualquier tipo de tablas).

Acciones

Se utiliza una clase por acción, para esto el nombre de la clase debe ser nombreAction y el nombre del método es simplemente execute.

```
class listarAction extends sfAction
{
    public function execute($petición)
    {
        // ...
    }
}
```

2.3.3 Comentarios

Los ficheros fuentes siguen la convención del formato de documentación Doxygen. Esta documentación es generada por el sistema, lo cual permite extraerla directamente del código fuente, facilitando mantener la consistencia de la misma con el código fuente.

Formato de los comentarios en la implementación

Los ficheros fuentes pueden tener 4 estilos de comentarios: bloque, línea simple, de seguimiento y de fin de línea.

Bloque de comentarios

Un bloque de comentario es precedido por una línea en blanco para separarlo del resto del código. Los bloques de comentario tienen solo un asterisco al inicio de cada línea exceptuando la primera.

Ejemplo:

```
/**
 * Documentación Aquí.
 */
```

Comentarios de línea simple

Los comentarios cortos pueden aparecer en una sola línea alineados al mismo nivel que el código que lo sigue, además debe estar precedido de una línea en blanco. Si un comentario no puede ponerse en una línea simple entonces debe utilizarse un bloque de comentario.

Ejemplo:

```
if (condición) {  
/* Condición principal*/  
}
```

Comentarios de seguimiento

Comentarios muy cortos pueden aparecer al final de la línea de código que describen, pero están alejados lo suficiente para separarlo de las sentencias. Si más de un comentario de seguimiento aparece en un pedazo de código deben tener la misma alineación. Evitar el estilo de comentar cada línea de código que se usa en lenguaje ensamblador.

```
if (a == 2) {  
return TRUE; /* caso especial */  
} else {  
return isprime(a); /* solo funciona para los números primos */  
}
```

Comentarios de fin de línea

El delimitador de comentario // convierte en comentario una línea completa o una parte de una línea. No debe ser usado para hacer comentarios de varias líneas consecutivas; sin embargo, se usa en líneas consecutivas para comentar secciones de código.

2.4 Modelo de implementación

“El Modelo de Implementación describe cómo los elementos del modelo de diseño, como las clases, se implementan en términos de componentes, como ficheros de código fuente, ejecutables, etc., y cómo dependen los componentes unos de otros en los Diagramas de Componentes. El Modelo de Implementación describe también cómo se organizan los componentes en el entorno de implementación, es decir, cómo se distribuye el sistema asignando componentes ejecutables a nodos en el Diagrama de Despliegue.” (16)

El modelo de implementación describe los componentes a construir y su organización y dependencia entre los nodos físicos en los que funcionará la aplicación.

2.4.1 Diagrama de Despliegue



Figura 2.1. Diagrama de Despliegue

Descripción de la funcionalidad y capacidad de los nodos.

PC Cliente: Computadora cliente que se conecta a través de un navegador Web al servidor central donde reside la aplicación.

PC Servidor: Servidor central, que tiene instalado un servidor Web, un servidor de base de datos y hospeda todos los componentes necesarios para el funcionamiento del producto.

Descripción de elementos e interfaces de comunicación

<<HTTP>>

Representa la conexión que se establece entre una computadora cliente con el servidor central donde reside el servidor Web y el de base de datos, en otras palabras, significa la conexión entre el navegador Web y el servidor de aplicaciones.

2.4.2 Diagrama de Componentes

Los Diagramas de Componentes muestran las relaciones que existen entre los componentes software de un sistema, estos representan todos los tipos de elementos software que se usan en la construcción de aplicaciones informáticas, como paquetes, bibliotecas, archivos de código fuente, binario y ejecutable. Este diagrama se utiliza para representar una vista estática de cualquier sistema.

A continuación se muestra el Diagrama de Componentes general de la aplicación que permite visualizar los cuestionarios interactivos y la interacción con los mismos. (*frontend*).

Capítulo 2: Implementación

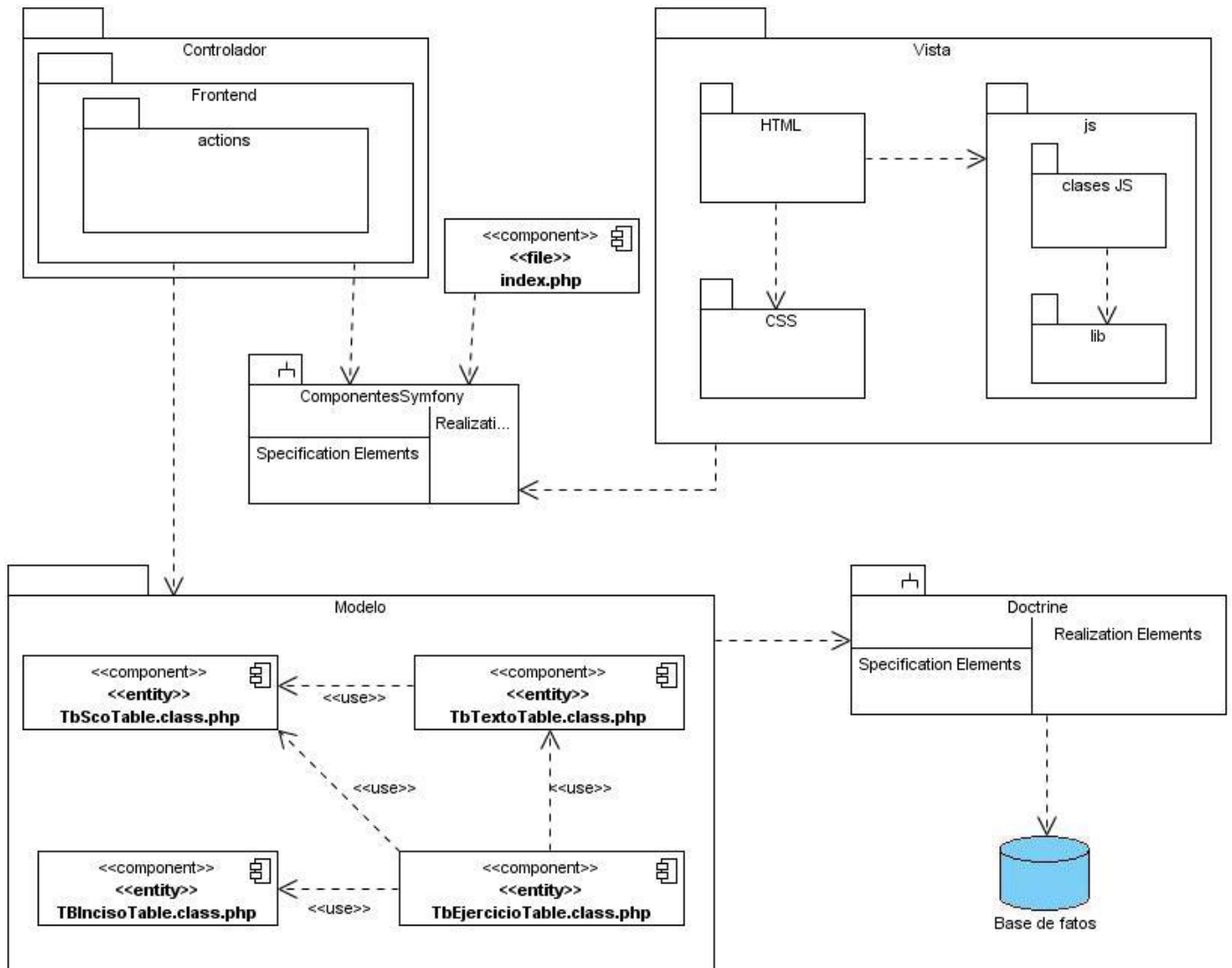


Figura 2.2. Diagrama de Componentes general del *frontend*

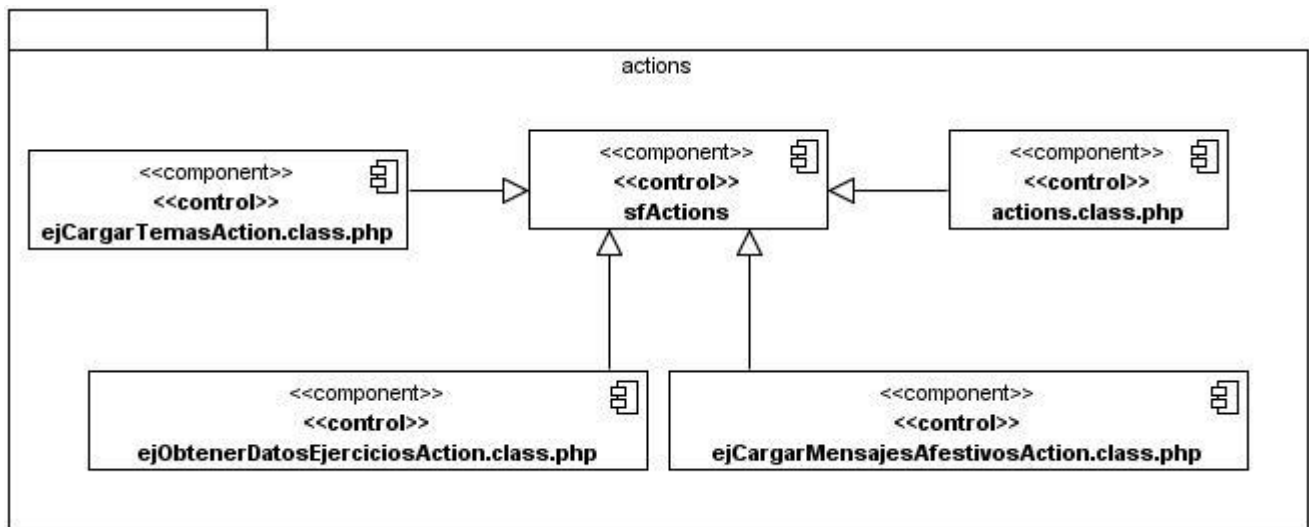


Figura 2.3. Diagrama de Componentes del paquete actions del *frontend*

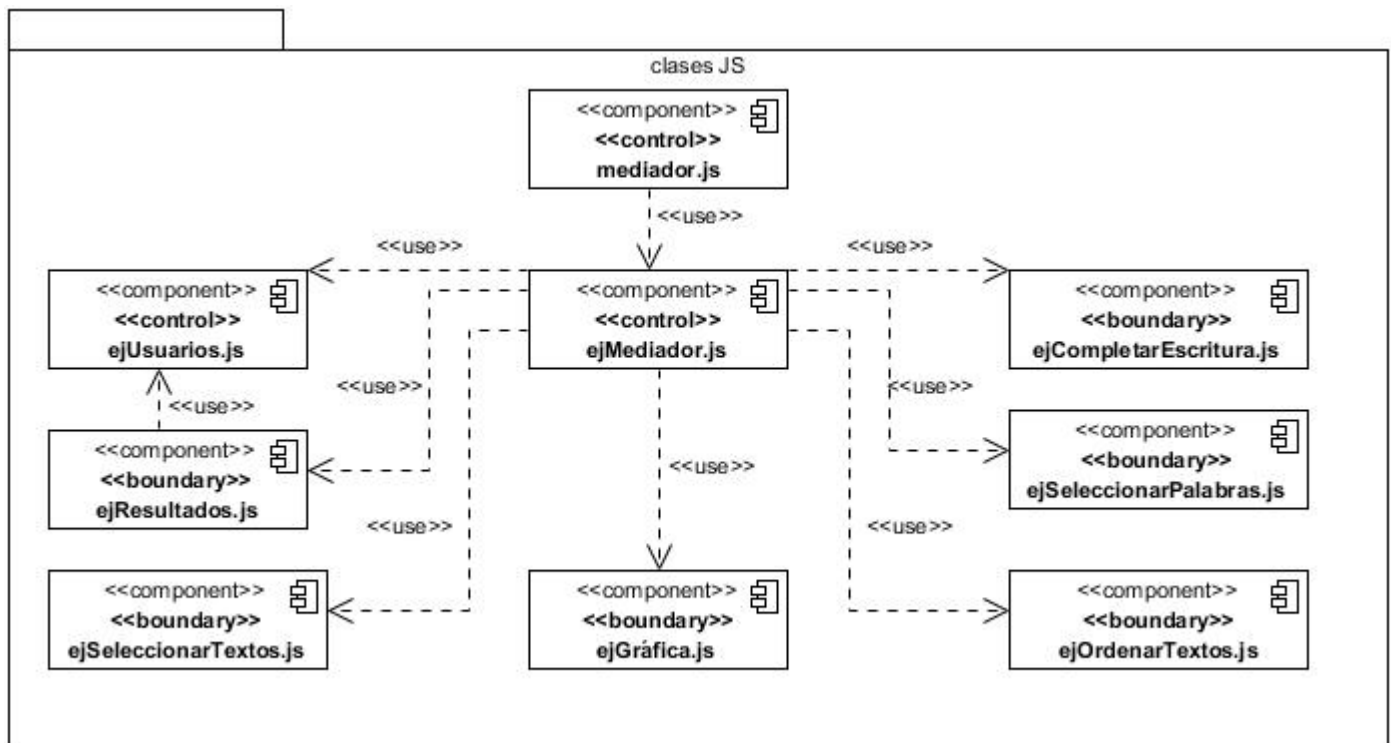


Figura 2.4. Diagrama de Componentes del paquete clases JS del *frontend*

Capítulo 2: Implementación

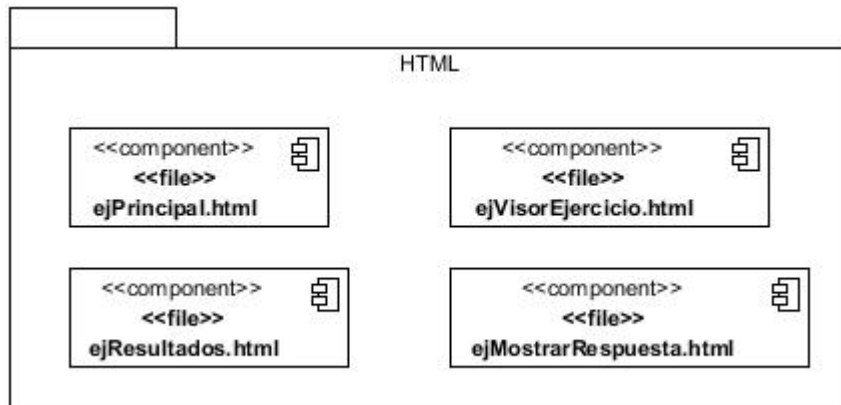


Figura 2.5 Diagrama de Componentes del paquete HTML del *frontend*

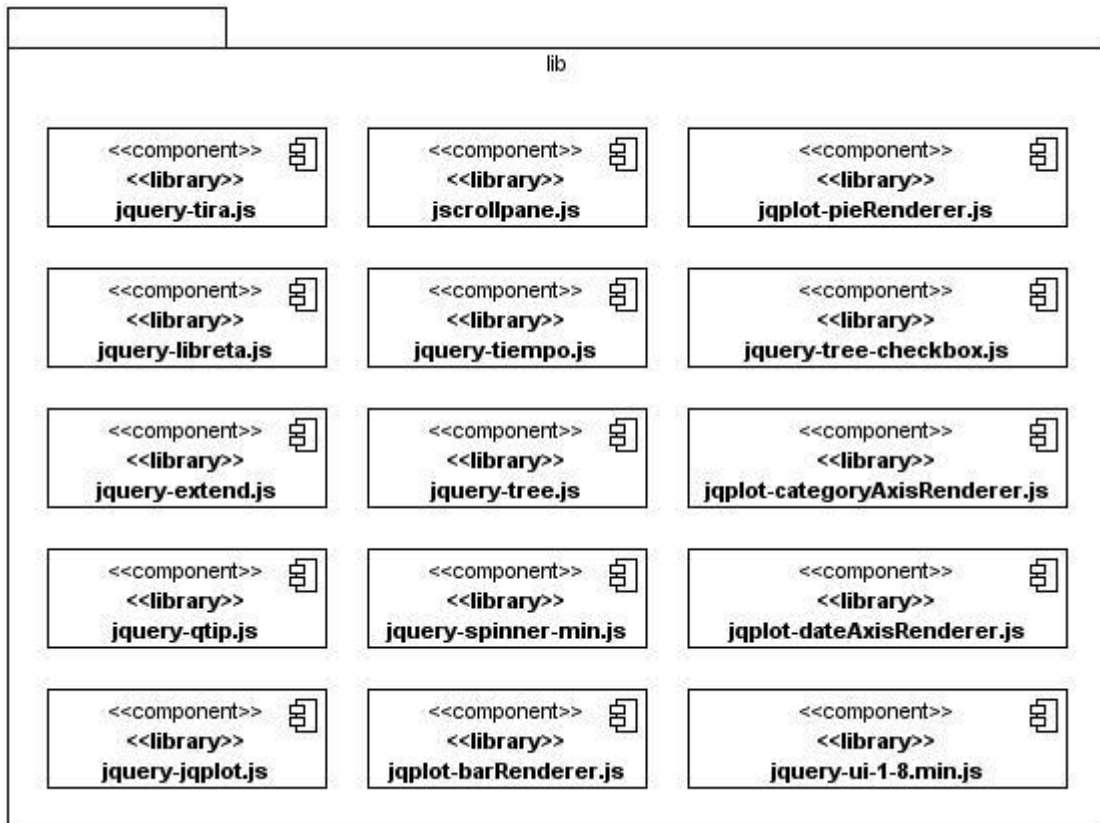


Figura 2.6. Diagrama de Componentes del paquete lib del *frontend*

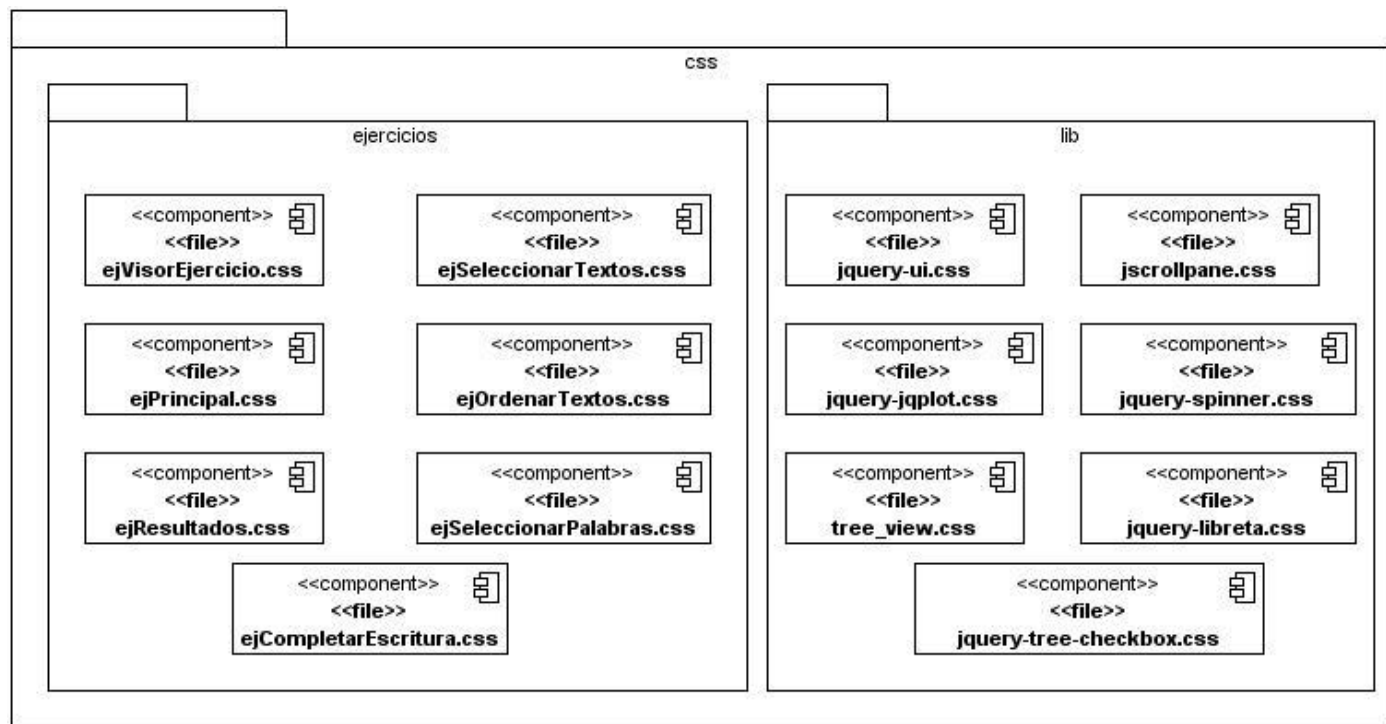


Figura 2.7. Diagrama de Componentes del paquete CSS del *frontend*

A continuación se muestra el Diagrama de Componentes general de la herramienta encargada de gestionar los contenidos educativos de los cuestionarios interactivos (*backend*).

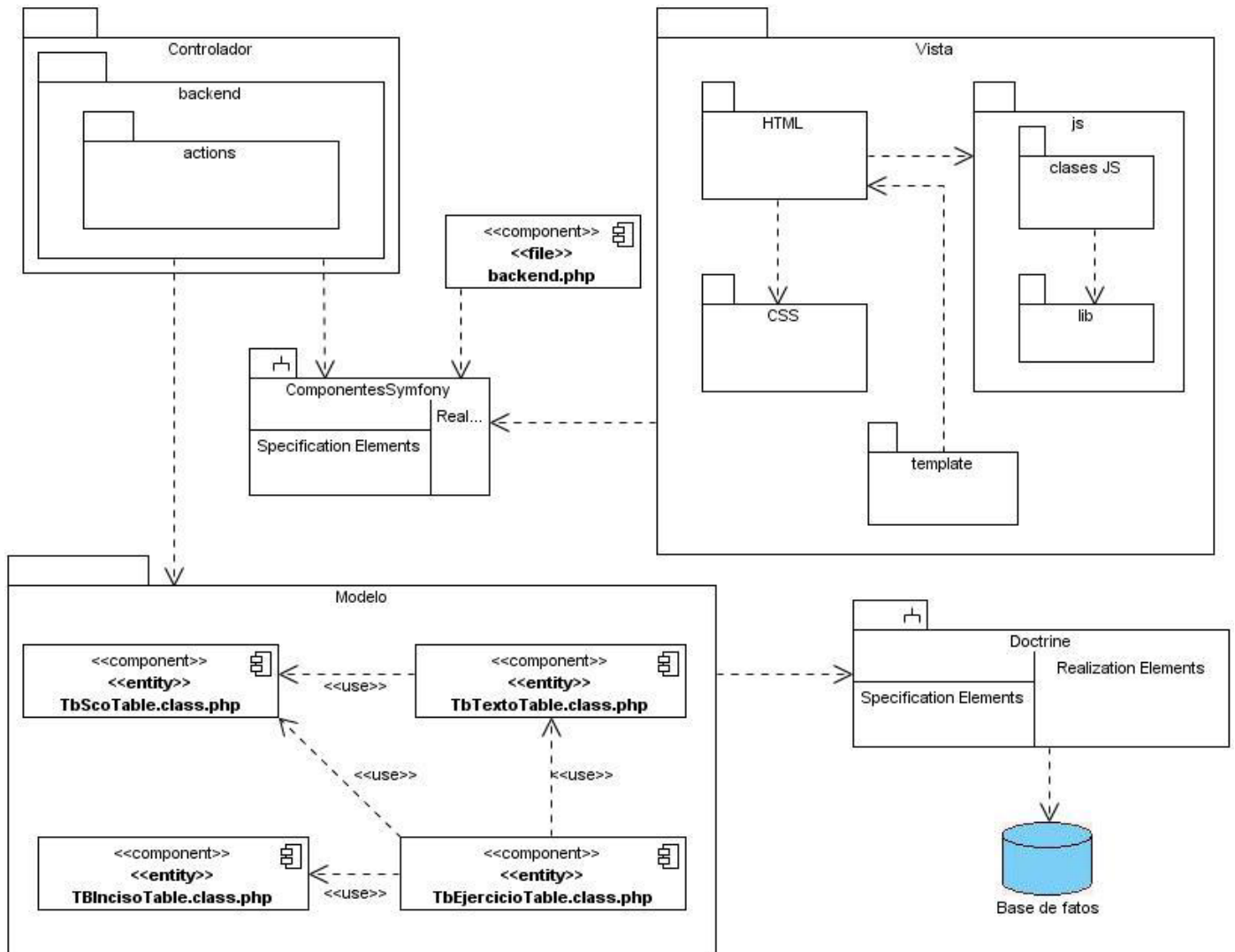


Figura 2.8. Diagrama de Componentes general del *backend*

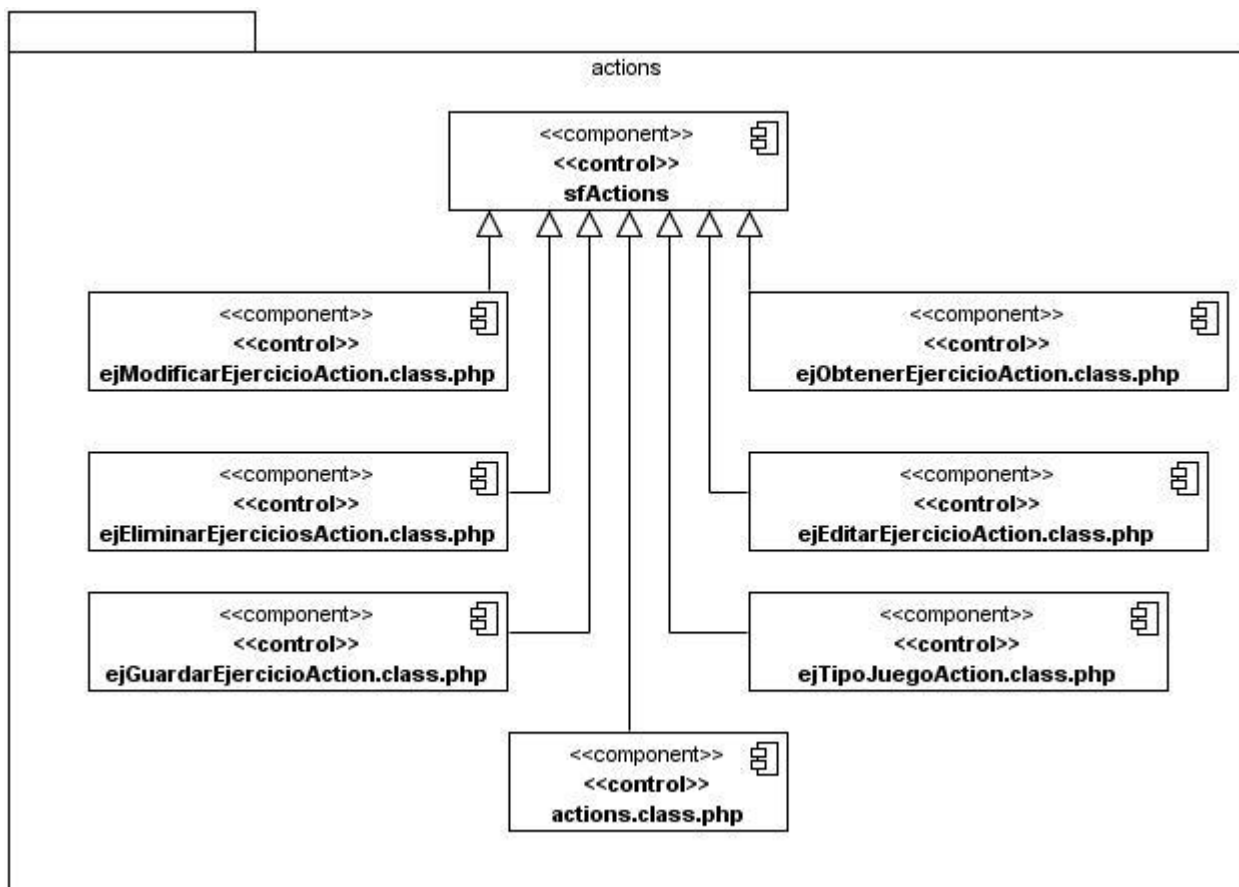


Figura 2.9. Diagrama de Componentes del paquete actions del *backend*

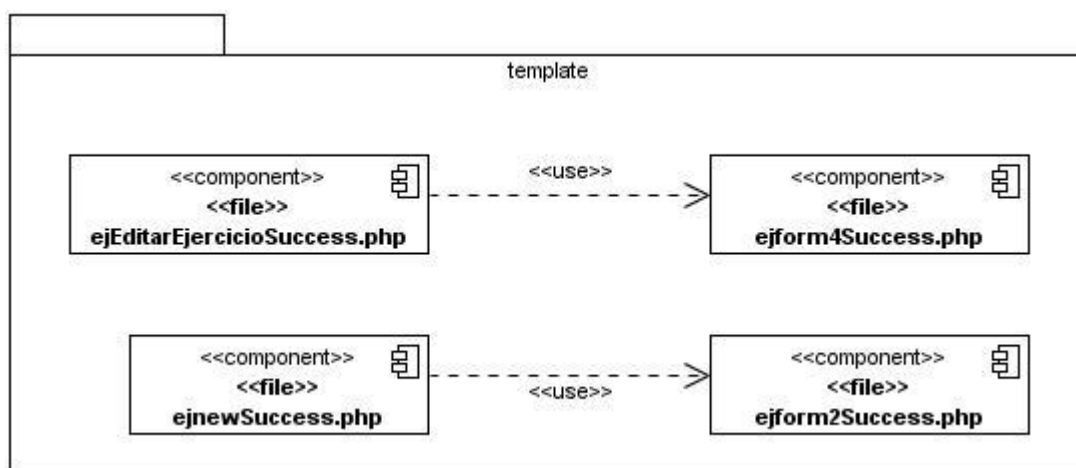


Figura 2.10. Diagrama de Componentes del paquete template del *backend*

Capítulo 2: Implementación

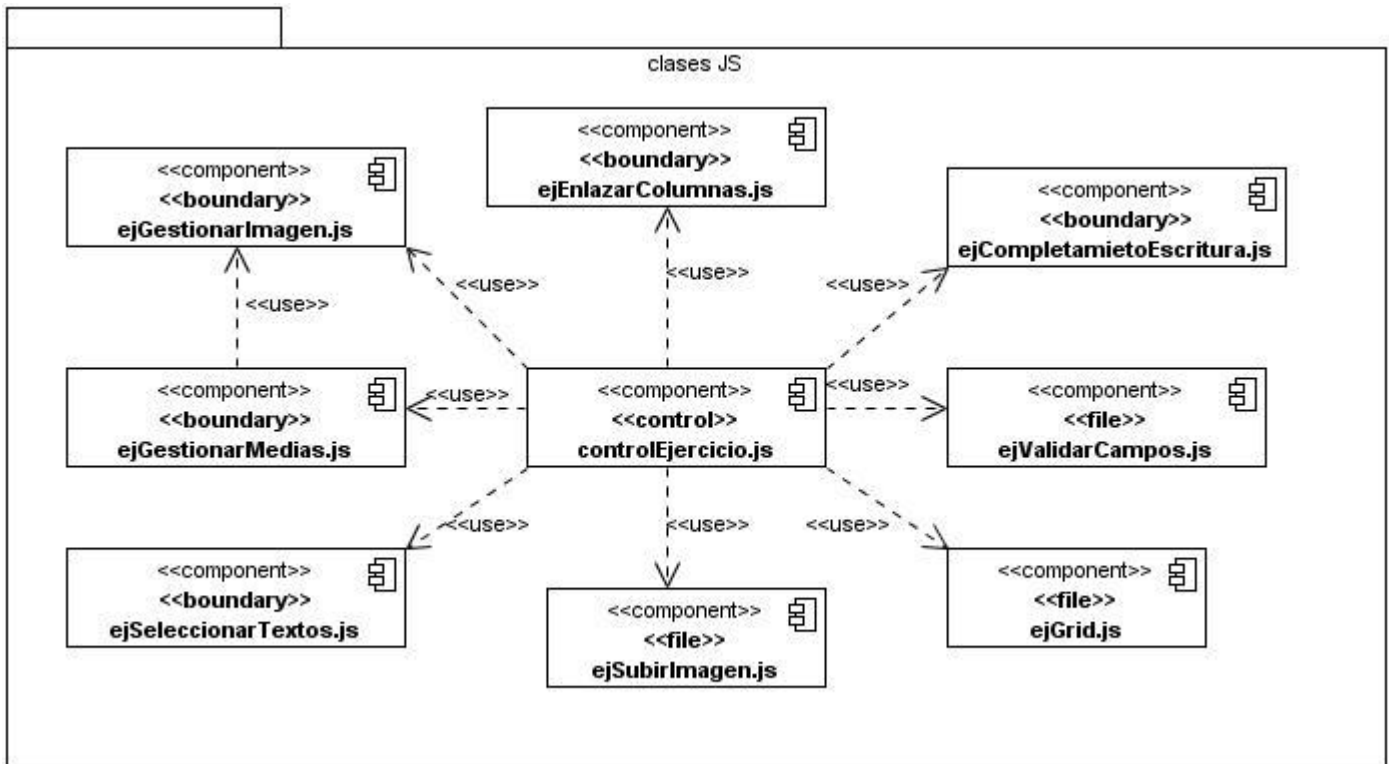


Figura 2.11. Diagrama de Componentes del paquete clases JS del *backend*

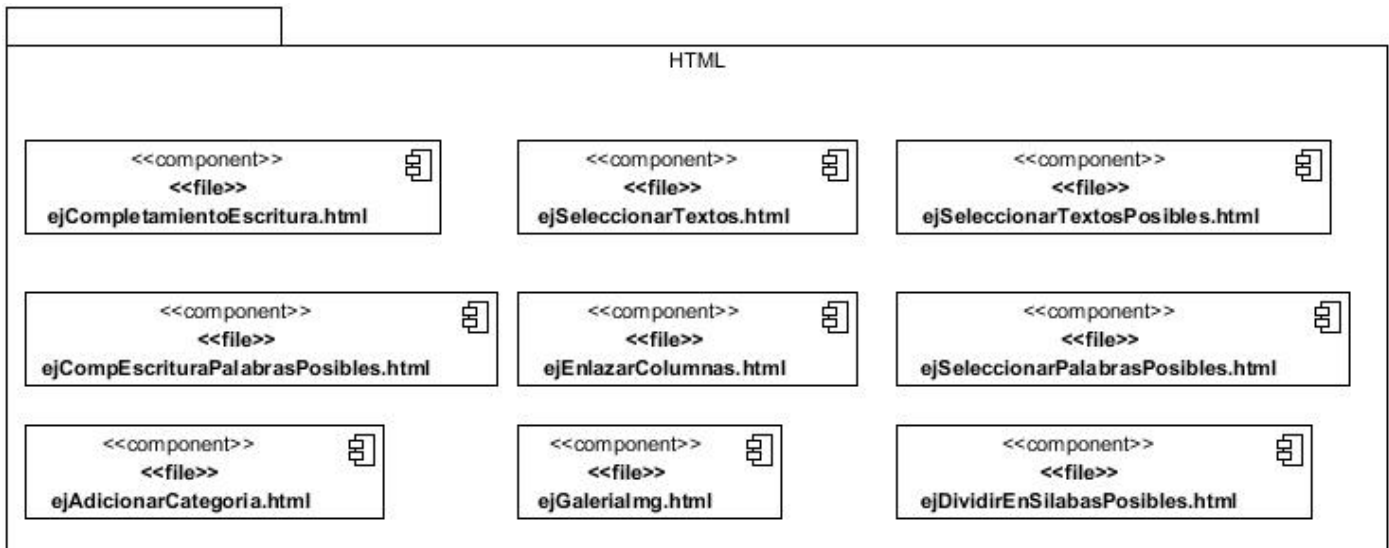


Figura 2.12. Diagrama de Componentes del paquete HTML del *backend*

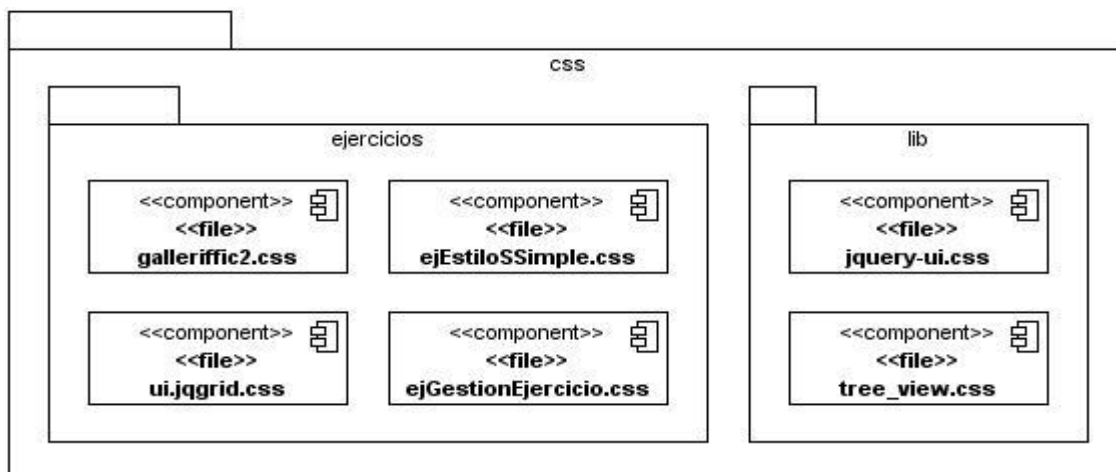


Figura 2.13. Diagrama de Componentes del paquete CSS del *backend*

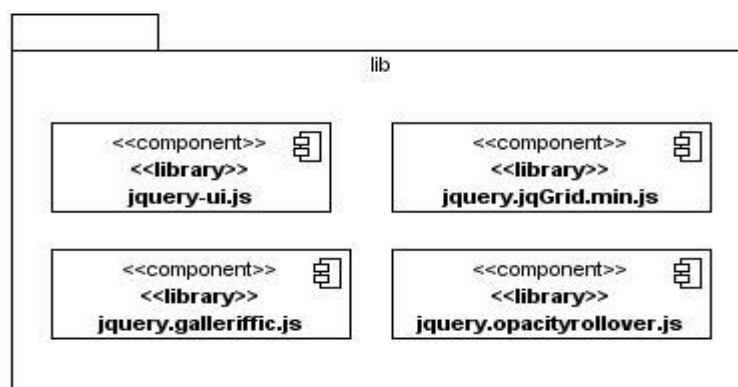


Figura 2.14. Diagrama de Componentes del paquete lib del *backend*

2.5 Conclusiones

En el presente capítulo se presentaron los Diagramas de Componentes elaborados, los cuales permitieron obtener un mejor entendimiento de la descomposición física del sistema en componentes, a efectos de construcción y funcionamiento, además el Diagrama de Despliegue permitió definir los nodos que se utilizarán para la implantación de la aplicación y se describió la estandarización del código generado en el módulo Ejercicios.

Capítulo 3

PRUEBAS

Las pruebas son un conjunto de actividades en las cuales un sistema es ejecutado bajo condiciones o requerimientos especificados, donde los resultados son observados y registrados para dar una evaluación de algún aspecto del sistema y determinar así la calidad del mismo. Tienen como objetivo validar que todos los requerimientos hayan sido implementados correctamente. Una prueba puede probar la presencia de errores pero no la ausencia de ellos. (36)

En el presente capítulo se caracterizan la técnica y el método de prueba a utilizar. Se definen y describen los casos de prueba diseñados con el objetivo de encontrar la mayor cantidad de fallos en el sistema y corregirlos, garantizando de esta forma la calidad del sistema y el total cumplimiento de los requisitos establecidos con el cliente. Se presentan los resultados obtenidos al aplicar las pruebas.

3.1 Técnicas de prueba

Las técnicas de prueba proporcionan distintos criterios para generar casos de prueba que provoquen fallos en los programas. Estas técnicas se agrupan en: (37)

- Técnicas de caja blanca o estructurales, que se basan en un minucioso examen de los detalles procedimentales del código a evaluar, por lo que es necesario conocer la lógica del programa.
- Técnicas de caja negra o funcionales, que realizan pruebas sobre la interfaz del programa a probar, entendiendo por interfaz las entradas y salidas de dicho programa. No es necesario conocer la lógica del programa, únicamente la funcionalidad que debe realizar.

Teniendo en cuenta que las pruebas de caja negra se centran en las funciones, las entradas y las salidas y que se necesita asegurar un buen funcionamiento de los requisitos funcionales, incluyendo la navegación, entrada de datos y obtención de resultados, en el presente trabajo se aplicarán pruebas funcionales.

3.1.1 Diseño de casos de prueba

Un caso de prueba especifica una forma de probar el sistema, incluyendo la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse. (17)

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 externo del sistema, define cómo probar un caso de uso o un escenario específico del mismo, incluye la verificación del resultado de la interacción entre los actores y el sistema, que se satisfacen las precondiciones y postcondiciones precisadas por el caso de uso. (17)

Para confeccionar los casos de prueba de caja negra existen distintos criterios o métodos, entre ellos se encuentran partición equivalente y análisis de valores límites, la variante que se seleccionó para realizar los casos de prueba es partición equivalente, cuyo objetivo fundamental es dividir el dominio de entrada de un programa en clases de datos, a partir de las cuales se derivan los casos de prueba. (37)

El diseño de casos de prueba según el método de partición equivalente consta de dos pasos: (37)

- Identificar las clases de equivalencia: Una clase de equivalencia representa un conjunto de estados válidos y no válidos para las condiciones de entrada de un programa.
- Identificar los casos de prueba: Para crear los casos de prueba a partir de las clases de equivalencia se han de seguir los siguientes pasos:
 1. Asignar a cada clase de equivalencia un número único.
 2. Hasta que todas las clases de equivalencia hayan sido cubiertas por los casos de prueba, se escribe un caso que cubra tantas clases válidas no incorporadas como sea posible.
 3. Hasta que todas las clases de equivalencia no válidas hayan sido cubiertas por casos de prueba, se escribe un caso de prueba que cubra una única clase no válida no cubierta.

A continuación se muestra el diseño de caso de prueba del caso de uso Realizar ejercicios, los restantes se encuentran en el Anexo 2.

3.1.1.1 Diseño de caso de prueba del CU Realizar ejercicios.

Descripción general

El caso de uso se inicia cuando el usuario accede al módulo Ejercicios. El sistema permite seleccionar los temas, el tipo de selección de ejercicios (Secuencial, Al azar o Asignado) y la cantidad de ejercicios a realizar. El usuario selecciona los temas, el tipo de selección de ejercicios, la cantidad de ejercicios y la opción comenzar. El sistema muestra los ejercicios, el usuario realiza los mismos y selecciona la opción que permite revisar su respuesta; al concluir el último intento de cada ejercicio el sistema permite ver la respuesta correcta del ejercicio y la respuesta del usuario en todos los intentos. Además permite ver las estadísticas de la ejercitación en cualquier momento. El usuario selecciona una de estas opciones si así lo desea, el sistema muestra el resultado correspondiente a la opción en cuestión y finaliza así el caso de uso.

Condiciones de ejecución

Debe haberse mostrado el escritorio de trabajo del usuario autenticado.

Secciones a probar en el caso de uso

Nombre de la sección	Escenarios de la sección	Descripción de la funcionalidad	Flujo central
SC 1: Realizar ejercicios “Secuencial”.	EC 1.1: Realizar ejercicios “Secuencial” exitosamente.	Muestra la pantalla principal del módulo Ejercicios. Muestra: <ul style="list-style-type: none"> • Por niveles los temas con ejercicios disponibles. • El tipo de selección de ejercicios “Secuencial” de forma predeterminada. Permite seleccionar uno o varios temas. El usuario selecciona uno o varios temas y se muestra en el campo desde del tipo de selección de ejercicios Secuencial el valor 1 y en el campo hasta la cantidad total de ejercicios disponibles en los temas seleccionados. Permite:	Autenticar usuario / Ejercicios / Comenzar / Revisa tu respuesta / Terminar

		<ul style="list-style-type: none">• Comenzar a visualizar ejercicios.• Seleccionar los valores desde y hasta.• Introducir los valores desde y hasta.• Seleccionar el tipo de selección de ejercicios “Al azar”. Ver SC 2: “Realizar ejercicios “Al azar”.• Seleccionar el tipo de selección de ejercicios “Asignados”. Ver SC 3: Realizar ejercicios “Asignados”.• Reiniciar la configuración de selección de ejercicios. <p>Luego el usuario introduce el rango de ejercicios que desea realizar de forma secuencial, en el campo desde el ejercicio inicial y en el campo hasta el ejercicio final</p> <p>A continuación selecciona la opción Comenzar.</p> <p>El sistema registra la traza de la interacción del usuario con el módulo. Ver caso de prueba del CU Registrar traza, carga una tipología de ejercicio y muestra los siguientes datos:</p> <ul style="list-style-type: none">• Enunciado.• Pregunta.• Intento.• Tiempo.• Número del ejercicio y el tema al que pertenece.• El nombre del usuario que está realizando el ejercicio. <p>Permite:</p>	
--	--	--	--

		<ul style="list-style-type: none">• Realizar el ejercicio.• Ver el ejercicio siguiente.• Ver el ejercicio anterior.• Revisar la respuesta.• Ver las estadísticas de la ejercitación.• Terminar la ejercitación. <p>El usuario realiza el ejercicio visualizado y selecciona la opción Revisa tu respuesta.</p> <p>Posteriormente el sistema ejecuta las siguientes acciones:</p> <ul style="list-style-type: none">• Registra la traza de la interacción del usuario con el sistema. <i>Ver caso de prueba del CU Registrar traza.</i>• Muestra un mensaje informando que la respuesta es incorrecta y un mensaje afectivo para alentar al usuario en la resolución de ejercicios.• Reproduce en voz de la mascota el mensaje afectivo mostrado y permite interactuar con la mascota. <i>Ver caso de prueba del CU Manipular mascota del módulo General.</i>• Muestra un mensaje conclusivo sobre la respuesta del ejercicio.• Permite ver la respuesta correcta del ejercicio realizado y la respuesta del usuario en cada uno de los intentos.• Permite terminar la ejercitación.• Bloquea el permiso de edición sobre el ejercicio.	
--	--	---	--

	<p>El usuario selecciona la opción Terminar.</p> <p>El sistema registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba del CU Registrar traza</i> y finaliza la ejercitación.</p>	
<p>EC 1.2: El número introducido en el campo desde es mayor que la cantidad de ejercicios disponibles.</p>	<p>Señala el campo desde como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje: “Este valor está fuera del intervalo”.</p> <p>Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios</p>
<p>EC 1.3: El campo desde y/o hasta está vacío.</p>	<p>Señala el campo desde y/o hasta como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”.</p> <p>Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios</p>
<p>EC 1.4: El número introducido en el campo hasta es menor que el número introducido en el campo desde.</p>	<p>Señala el campo hasta como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”. Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios</p>
<p>EC 1.5: Los valores introducidos en los campos desde y/o hasta no son números enteros.</p>	<p>Señala el campo desde y/o hasta como incorrectos y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”. Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios</p>
<p>EC 1.6: No modifica los valores (desde - hasta) de la cantidad de ejercicios a realizar.</p>	<p>Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios</p>

Capítulo 3: Pruebas

	EC 1.7: Selecciona los valores del rango (desde – hasta) de ejercicios a visualizar.	Luego regresa al EC 1.1.	Autenticar usuario / Ejercicios
	EC 1.8: Selecciona la opción Configuración predeterminada.	Reinicia la configuración del módulo. Luego regresa al EC 1.1.	Autenticar usuario / Ejercicios / Configuración predeterminada
	EC 1.9: La cantidad de ejercicios seleccionada no es suficiente para la cantidad de estudiantes que iniciaron sesión.	Muestra el siguiente mensaje: “La cantidad de ejercicios seleccionada no es suficiente para todos los estudiantes autenticados en la sesión”. Luego regresa al EC 1.1.	Autenticar usuario / Ejercicios / Comenzar
	EC 1.10: Se autenticó más de un estudiante en el sistema.	Muestra una pantalla con los nombres de los estudiantes autenticados en el orden en el que cada uno realizará ejercicios. Permite Aceptar o Cancelar. El usuario selecciona la opción Aceptar. Regresa al EC 1.1.	Autenticar usuario / Ejercicios / Comenzar / Aceptar
	EC 1.11: Selecciona la opción de ver el ejercicio siguiente y este no ha sido resuelto.	Muestra secuencialmente el ejercicio siguiente. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza.</i> Luego regresa al EC 1.1.	Autenticar usuario / Ejercicios / Comenzar / Siguiente

Capítulo 3: Pruebas

	<p>EC 1.12: Selecciona la opción de ver el ejercicio siguiente y este es el último.</p>	<p>Muestra secuencialmente el ejercicio siguiente. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza.</i> Muestra la opción siguiente deshabilitada. Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Siguiente</p>
	<p>EC 1.13: Selecciona la opción de ver el ejercicio anterior y este no ha sido resuelto.</p>	<p>Muestra secuencialmente el ejercicio anterior. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza.</i> Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Anterior</p>
	<p>EC 1.14: Selecciona la opción de ver el ejercicio anterior y este es el primero.</p>	<p>Muestra secuencialmente el ejercicio anterior. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza.</i> Muestra la opción siguiente deshabilitada. Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Anterior</p>
	<p>EC 1.15: Selecciona la opción de ver el ejercicio anterior y este ya fue resuelto.</p>	<p>Muestra el ejercicio anterior con las respuestas dadas por el usuario. Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Anterior</p>
	<p>EC 1.16: Selecciona la opción de ver el ejercicio siguiente y este ya fue resuelto.</p>	<p>Muestra el ejercicio siguiente con las respuestas dadas por el usuario. Luego regresa al EC 1.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Siguiente</p>
	<p>EC 1.17: Selecciona la opción de ver las estadísticas de la</p>	<p>Muestra:</p> <ul style="list-style-type: none"> • El porcentaje de efectividad obtenido en la realización de los ejercicios. 	<p>Autenticar usuario / Ejercicios /</p>

	ejercitación.	<ul style="list-style-type: none"> • La cantidad de ejercicios sin responder. • Una gráfica de barras con la cantidad de ejercicios por cada categoría de calificación. <p>Luego regresa al EC 1.1.</p>	Comenzar / Estadísticas
	EC 1.18: La respuesta dada al ejercicio es correcta.	<p>Muestra un mensaje informando que la respuesta es correcta.</p> <p>Muestra un mensaje afectivo felicitando al usuario por su respuesta.</p> <p>Reproduce en voz de la mascota el mensaje afectivo mostrado.</p> <p>Permite interactuar con la mascota. Ver CU Manipular mascota del módulo General. Luego regresa al EC 1.1.</p>	Autenticar usuario / Ejercicios / Comenzar / Revisa tu respuesta
	EC 1.19: Quedan intentos disponibles para realizar el ejercicio.	<p>Muestra un mensaje informando que la respuesta es incorrecta.</p> <p>Muestra un mensaje afectivo para alentar al usuario en la resolución del ejercicio.</p> <p>Reproduce en voz de la mascota el mensaje afectivo mostrado y permite interactuar con la mascota. Ver CU Manipular mascota del módulo General.</p> <p>Muestra un mensaje con una pista de ayuda.</p> <p>Permite realizar nuevamente el ejercicio. Luego regresa al EC 1.1.</p>	Autenticar usuario / Ejercicios / Comenzar / Revisa tu respuesta
	EC 1.20: Selecciona la opción Respuesta correcta.	Muestra la pantalla Comparación de resultados, en la columna izquierda se muestra la respuesta correcta del ejercicio y en la columna derecha la respuesta del usuario en cada uno de los intentos.	Autenticar usuario / Ejercicios / Comenzar /

		<p>Permite seleccionar la opción Aceptar para salir de la pantalla.</p> <p>El usuario selecciona la opción Aceptar.</p> <p>El sistema regresa a la vista anterior. Luego regresa al EC 1.1.</p>	<p>Respuesta correcta</p>
<p>SC 2: Realizar ejercicios “Al azar”</p>	<p>EC 2.1: Realizar ejercicios “Al azar” exitosamente</p>	<p>Muestra la cantidad total de ejercicios disponibles de los temas seleccionados en el campo cantidad.</p> <p>Permite:</p> <ul style="list-style-type: none"> • Introducir el número de ejercicios que desea visualizar. • Seleccionar la cantidad de ejercicios a visualizar. • Comenzar la visualización de ejercicios. <p>El usuario introduce la cantidad de ejercicios.</p> <p>A continuación selecciona la opción Comenzar.</p> <p>El sistema registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba del CU Registrar traza</i>, carga una tipología de ejercicio y muestra los siguientes datos:</p> <ul style="list-style-type: none"> • Enunciado. • Pregunta. • Intento. • Tiempo. • Número del ejercicio y el tema al que pertenece. • El nombre del usuario que está realizando el ejercicio. <p>Permite:</p> <ul style="list-style-type: none"> • Realizar el ejercicio. 	<p>Autenticar usuario / Ejercicios / Comenzar / Revisa tu respuesta / Terminar</p>

		<ul style="list-style-type: none">• Ver el ejercicio siguiente.• Ver el ejercicio anterior.• Revisar la respuesta.• Ver las estadísticas de la ejercitación.• Terminar la ejercitación. <p>El usuario realiza el ejercicio visualizado y selecciona la opción Revisa tu respuesta.</p> <p>Posteriormente el sistema ejecuta las siguientes acciones:</p> <ul style="list-style-type: none">• Registra la traza de la interacción del usuario con el sistema. <i>Ver caso de prueba del CU Registrar traza.</i>• Muestra un mensaje informando que la respuesta es incorrecta y un mensaje afectivo para alentar al usuario en la resolución de ejercicios.• Reproduce en voz de la mascota el mensaje afectivo mostrado y permite interactuar con la mascota. <i>Ver caso de prueba del CU Manipular mascota del módulo General.</i>• Muestra un mensaje conclusivo sobre la respuesta del ejercicio.• Permite ver la respuesta correcta del ejercicio realizado y la respuesta del usuario en cada uno de los intentos.• Permite terminar la ejercitación.• Bloquea el permiso de edición sobre el ejercicio. <p>El usuario selecciona la opción Terminar.</p>	
--	--	---	--

		El sistema registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba del CU Registrar traza</i> y finaliza la ejercitación.	
	EC 2.2: El número introducido en el campo “cantidad” es mayor que la cantidad de ejercicios disponibles.	Señala el campo cantidad como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”. Luego regresa al escenario 2.1.	Autenticar usuario / Ejercicios / Comenzar
	EC 2.3: El campo cantidad está vacío.	Señala el campo cantidad como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”. Luego regresa al escenario 2.1.	Autenticar usuario / Ejercicios / Comenzar
	EC 2.4: Selecciona un cantidad específica del total mostrado.	Regresa al escenario 2.1.	Autenticar usuario / Ejercicios / Comenzar
	EC 2.5: El número introducido en el campo “cantidad” es mayor que la cantidad de ejercicios disponibles.	Señala el campo cantidad como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”. Luego regresa al escenario 2.1.	Autenticar usuario / Ejercicios / Comenzar / Revisa tu respuesta / Terminar
	EC 2.6: La cantidad de ejercicios seleccionada no es suficiente para la cantidad de	Muestra el siguiente mensaje: “La cantidad de ejercicios seleccionada no es suficiente para todos los estudiantes autenticados en la sesión”. Luego regresa al EC 2.1.	Autenticar usuario / Ejercicios / Comenzar

Capítulo 3: Pruebas

	estudiantes que iniciaron sesión.		
	EC 2.7: Se autenticó más de un estudiante en el sistema.	Muestra una pantalla con los nombres de los estudiantes autenticados en el orden en el que cada uno realizará ejercicios. Permite Aceptar o Cancelar. El usuario selecciona la opción Aceptar. Regresa al EC 2.1.	Autenticar usuario / Ejercicios / Comenzar / Aceptar
	EC 2.8: Selecciona la opción de ver el ejercicio siguiente y este no ha sido resuelto.	Muestra secuencialmente el ejercicio siguiente. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza</i> . Luego regresa al EC 2.1.	Autenticar usuario / Ejercicios / Comenzar / Siguiente
	EC 2.9: Selecciona la opción de ver el ejercicio siguiente y este es el último.	Muestra secuencialmente el ejercicio siguiente. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza</i> . Muestra la opción siguiente deshabilitada. Luego regresa al EC 2.1.	Autenticar usuario / Ejercicios / Comenzar / Siguiente
	EC 2.10: Selecciona la opción de ver el ejercicio anterior y este no ha sido resuelto.	Muestra secuencialmente el ejercicio anterior. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza</i> . Luego regresa al EC 2.1.	Autenticar usuario / Ejercicios / Comenzar / Anterior
	EC 2.11: Selecciona la opción de ver el ejercicio anterior y este es el primero.	Muestra secuencialmente el ejercicio anterior. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza</i> . Muestra la opción siguiente deshabilitada. Luego regresa al EC 2.1.	Autenticar usuario / Ejercicios / Comenzar / Anterior

	<p>EC 2.12: Selecciona la opción de ver el ejercicio anterior y este ya fue resuelto.</p>	<p>Muestra el ejercicio anterior con las respuestas dadas por el usuario. Luego regresa al EC 2.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Anterior</p>
	<p>EC 2.13: Selecciona la opción de ver el ejercicio siguiente y este ya fue resuelto.</p>	<p>Muestra el ejercicio siguiente con las respuestas dadas por el usuario. Luego regresa al EC 2.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Siguiente</p>
	<p>EC 2.14: Selecciona la opción de ver las estadísticas de la ejercitación.</p>	<p>Muestra:</p> <ul style="list-style-type: none"> • El porcentaje de efectividad obtenido en la realización de los ejercicios. • La cantidad de ejercicios sin responder. • Una gráfica de barras con la cantidad de ejercicios por cada categoría de calificación. <p>Luego regresa al EC 2.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Estadísticas</p>
	<p>EC 2.15: La respuesta dada al ejercicio es correcta.</p>	<p>Muestra un mensaje informando que la respuesta es correcta.</p> <p>Muestra un mensaje afectivo felicitando al usuario por su respuesta.</p> <p>Reproduce en voz de la mascota el mensaje afectivo mostrado.</p> <p>Permite interactuar con la mascota. Ver CU Manipular mascota del módulo General. Luego regresa al EC 2.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Revisa tu respuesta</p>
	<p>EC 2.16: Quedan intentos disponibles</p>	<p>Muestra un mensaje informando que la respuesta es incorrecta.</p>	<p>Autenticar usuario /</p>

Capítulo 3: Pruebas

	<p>para realizar el ejercicio.</p>	<p>Muestra un mensaje afectivo para alentar al usuario en la resolución del ejercicio.</p> <p>Reproduce en voz de la mascota el mensaje afectivo mostrado y permite interactuar con la mascota. Ver CU Manipular mascota del módulo General.</p> <p>Muestra un mensaje con una pista de ayuda.</p> <p>Permite realizar nuevamente el ejercicio. Luego regresa al EC 2.1.</p>	<p>Ejercicios / Comenzar / Revisa tu respuesta</p>
	<p>EC 2.17: Selecciona la opción Respuesta correcta.</p>	<p>Muestra la pantalla Comparación de resultados, en la columna izquierda se muestra la respuesta correcta del ejercicio y en la columna derecha la respuesta del usuario en cada uno de los intentos.</p> <p>Permite seleccionar la opción Aceptar para salir de la pantalla.</p> <p>El usuario selecciona la opción Aceptar.</p> <p>El sistema regresa a la vista anterior. Luego regresa al EC 2.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Respuesta correcta</p>
<p>SC 3: Realizar ejercicios “Asignados”</p>	<p>EC 2.1: Realizar ejercicios “Asignados” exitosamente</p>	<p>Permite seleccionar la forma de asignar los ejercicios: Individual y En equipo.</p> <p>El usuario selecciona la opción En equipo.</p> <p>El sistema permite visualizar los estudiantes que conformarán el equipo de trabajo.</p> <p>Muestra en un área de disponibles el número de cada tema seleccionado y los números de los ejercicios disponibles por tema.</p> <p>Brinda la posibilidad de seleccionar de cada tema el ejercicio o los ejercicios que serán asignados, mediante una opción que permite pasar ejercicios</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Revisa tu respuesta / Terminar</p>

		<p>del área de disponibles para un área de seleccionados.</p> <p>A continuación el usuario selecciona los ejercicios deseados en el área de disponibles y la opción que permite pasarlos para el área de seleccionados.</p> <p>El sistema elimina los ejercicios seleccionados del área de disponibles y los muestra en el área de disponibles, permite pasar ejercicios del área de seleccionados para el área de disponibles y comenzar la ejercitación.</p> <p>El usuario selecciona la opción Comenzar.</p> <p>El sistema registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba del CU Registrar traza.</i></p> <p>Muestra una pantalla con los nombres de los estudiantes autenticados en el orden en el que cada uno realizará ejercicios. Permite Aceptar.</p> <p>El usuario selecciona la opción Aceptar.</p> <p>Posteriormente el sistema carga una tipología de ejercicio y muestra los siguientes datos:</p> <ul style="list-style-type: none">• Enunciado.• Pregunta.• Intento.• Tiempo.• Número del ejercicio y el tema al que pertenece.• El nombre del usuario que está realizando el ejercicio. <p>Permite:</p>	
--	--	---	--

		<ul style="list-style-type: none">• Realizar el ejercicio.• Ver el ejercicio siguiente.• Ver el ejercicio anterior.• Revisar la respuesta.• Ver las estadísticas de la ejercitación.• Terminar la ejercitación. <p>El usuario realiza el ejercicio visualizado y selecciona la opción Revisa tu respuesta. Posteriormente el sistema ejecuta las siguientes acciones:</p> <ul style="list-style-type: none">• Registra la traza de la interacción del usuario con el sistema. <i>Ver caso de prueba del CU Registrar traza.</i>• Muestra un mensaje informando que la respuesta es incorrecta y un mensaje afectivo para alentar al usuario en la resolución de ejercicios.• Reproduce en voz de la mascota el mensaje afectivo mostrado y permite interactuar con la mascota. <i>Ver caso de prueba del CU Manipular mascota del módulo General.</i>• Muestra un mensaje conclusivo sobre la respuesta del ejercicio.• Permite ver la respuesta correcta del ejercicio realizado y la respuesta del usuario en cada uno de los intentos.• Permite terminar la ejercitación.• Bloquea el permiso de edición sobre el ejercicio.	
--	--	--	--

Capítulo 3: Pruebas

	El usuario selecciona la opción Terminar. El sistema registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba del CU Registrar traza</i> y finaliza la ejercitación.	
EC 3.2: Selecciona la opción Individual.	Permite seleccionar el nombre del usuario al que se asignarán los ejercicios. Selecciona el nombre del usuario al que se asignarán ejercicios. Luego regresa al escenario 3.1.	Autenticar usuario / Ejercicios
EC 3.3: Solo hay un estudiante autenticado en el sistema.	Muestra el siguiente mensaje: “No es posible crear un equipo de trabajo con un solo estudiante.” Luego regresa al escenario 3.1.	Autenticar usuario / Ejercicios
EC 3.4: La cantidad de ejercicios seleccionada es menor que la cantidad de estudiantes que forman el equipo de trabajo.	Muestra el siguiente mensaje “La cantidad de ejercicios seleccionada no es suficiente para todos los estudiantes del equipo de trabajo”. Regresa al escenario 3.1.	Autenticar usuario / Ejercicios / Comenzar
EC 3.5: Selecciona la opción de ver el ejercicio siguiente y este no ha sido resuelto.	Muestra secuencialmente el ejercicio siguiente. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza.</i> Luego regresa al EC 3.1.	Autenticar usuario / Ejercicios / Comenzar / Siguiente
EC 3.6: Selecciona la opción de ver el ejercicio siguiente y este es el último.	Muestra secuencialmente el ejercicio siguiente. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza.</i> Muestra la opción siguiente deshabilitada. Luego regresa al EC 3.1.	Autenticar usuario / Ejercicios / Comenzar / Siguiente

Capítulo 3: Pruebas

	<p>EC 3.7: Selecciona la opción de ver el ejercicio anterior y este no ha sido resuelto.</p>	<p>Muestra secuencialmente el ejercicio anterior. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza.</i> Luego regresa al EC 3.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Anterior</p>
	<p>EC 3.8: Selecciona la opción de ver el ejercicio anterior y este es el primero.</p>	<p>Muestra secuencialmente el ejercicio anterior. Registra la traza de la interacción del usuario con el módulo. <i>Ver caso de prueba CU Registrar traza.</i> Muestra la opción siguiente deshabilitada. Luego regresa al EC 3.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Anterior</p>
	<p>EC 3.9: Selecciona la opción de ver el ejercicio anterior y este ya fue resuelto.</p>	<p>Muestra el ejercicio anterior con las respuestas dadas por el usuario. Luego regresa al EC 3.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Anterior</p>
	<p>EC 3.10: Selecciona la opción de ver el ejercicio siguiente y este ya fue resuelto.</p>	<p>Muestra el ejercicio siguiente con las respuestas dadas por el usuario. Luego regresa al EC 3.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Siguiente</p>
	<p>EC 3.11: Selecciona la opción de ver las estadísticas de la ejercitación.</p>	<p>Muestra:</p> <ul style="list-style-type: none"> • El porcentaje de efectividad obtenido en la realización de los ejercicios. • La cantidad de ejercicios sin responder. • Una gráfica de barras con la cantidad de ejercicios por cada categoría de calificación. <p>Luego regresa al EC 3.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Estadísticas</p>
	<p>EC 3.12: La respuesta</p>	<p>Muestra un mensaje informando que la respuesta</p>	<p>Autenticar</p>

	<p>dada al ejercicio es correcta.</p>	<p>es correcta. Muestra un mensaje afectivo felicitando al usuario por su respuesta. Reproduce en voz de la mascota el mensaje afectivo mostrado. Permite interactuar con la mascota. Ver CU Manipular mascota del módulo General. Luego regresa al EC 3.1.</p>	<p>usuario / Ejercicios / Comenzar / Revisa tu respuesta</p>
	<p>EC 3.13: Quedan intentos disponibles para realizar el ejercicio.</p>	<p>Muestra un mensaje informando que la respuesta es incorrecta. Muestra un mensaje afectivo para alentar al usuario en la resolución del ejercicio. Reproduce en voz de la mascota el mensaje afectivo mostrado y permite interactuar con la mascota. Ver CU Manipular mascota del módulo General. Muestra un mensaje con una pista de ayuda. Permite realizar nuevamente el ejercicio. Luego regresa al EC 3.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Revisa tu respuesta</p>
	<p>EC 3.14: Selecciona la opción Respuesta correcta.</p>	<p>Muestra la pantalla Comparación de resultados, en la columna izquierda se muestra la respuesta correcta del ejercicio y en la columna derecha la respuesta del usuario en cada uno de los intentos. Permite seleccionar la opción Aceptar para salir de la pantalla. El usuario selecciona la opción Aceptar. El sistema regresa a la vista anterior. Luego regresa al EC 3.1.</p>	<p>Autenticar usuario / Ejercicios / Comenzar / Respuesta correcta</p>

Capítulo 3: Pruebas

	EC 3.15: Selecciona pasar ejercicios del área de seleccionados para el área de disponibles.	Elimina los ejercicios seleccionados del área de asignados y los muestra en el área de disponibles. Luego regresa al escenario 3.1.	Autenticar usuario / Ejercicios
--	--	---	---------------------------------

Tabla 3.1. Secciones a probar en el CU Realizar ejercicios

Descripción de variable

No	Nombre de campo	Clasificación	Valor Nulo	Descripción
1	Desde	Campo de texto y selección	Si	Se selecciona o introduce un número entero.
2	Hasta	Campo de texto y selección	Si	Se selecciona o introduce un número entero.
3	Cantidad	Campo de texto y selección	Si	Se selecciona o introduce un número entero.
6	Disponibles	Campo de selección	Si	Se muestran el número de cada tema seleccionado y los números de los
7	Seleccionados	Campo de selección	Si	Se muestran los ejercicios seleccionados de cada tema.

Tabla 3.2 Descripción de variables del caso de prueba del CU Realizar ejercicios.

Matriz de datos

Las celdas de las tablas que se muestran a continuación contienen las letras V, I o N/A. V indica válido, I indica inválido y N/A que no es necesario proporcionar un valor de la variable en este caso, ya que es irrelevante.

Capítulo 3: Pruebas

SC #1: Realizar ejercicios “Secuencial”.

ID del escenario	Escenario	Desde	Hasta	Respuesta del sistema	Resultado de la prueba
EC 1.1	Realizar ejercicios “Secuencial” exitosamente	V	V	Permite comenzar la ejercitación.	Permite comenzar la ejercitación.
EC 1.2	El número introducido en el campo desde es mayor que la cantidad de ejercicios disponibles.	I	V	Señala el campo desde como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje: “Este valor está fuera del intervalo”.	Señala el campo desde y muestra un <i>tooltip</i> con el mensaje.
EC 1.3	El campo desde y/o hasta está vacío.	I	V	Señala el campo desde como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje: “Este valor está fuera del intervalo”.	Señala el campo desde y muestra un <i>tooltip</i> con el mensaje.
		V	I	Señala el campo hasta como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje: “Este valor está fuera del intervalo”.	Señala el campo hasta y muestra un <i>tooltip</i> con el mensaje.
		I	I	Señala el campo desde y hasta como incorrecto y muestra sobre cada campo un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”.	Señala el campo desde y hasta, muestra sobre cada campo un <i>tooltip</i> con el mensaje.
EC 1.4	El número introducido en el campo hasta es menor que el número en el campo desde.	V	I	Señala el campo hasta como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”.	Señala el campo hasta y muestra un <i>tooltip</i> con el mensaje.

EC 1.5	Los valores introducidos en los campos desde y/o hasta no son números enteros.	I	I	Señala el campo desde y hasta como incorrecto y muestra sobre cada campo un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”.	Señala el campo desde y hasta, muestra sobre cada campo un <i>tooltip</i> con el mensaje.
		V	I	Señala el campo hasta como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”.	Señala el campo hasta y muestra un <i>tooltip</i> con el mensaje.
		I	V	Señala el campo desde como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”.	Señala el campo desde y muestra un <i>tooltip</i> con el mensaje.
EC 1.9	La cantidad de ejercicios seleccionada no es suficiente para la cantidad de estudiantes que iniciaron sesión.	V	I	Muestra el siguiente mensaje: “La cantidad de ejercicios seleccionada no es suficiente para todos los estudiantes autenticados en la sesión”.	Muestra el mensaje.
		I	V		
		I	I		

Tabla 3.3 Matriz de datos caso de prueba del CU Realizar ejercicios.

SC #2: Realizar ejercicios “Al azar”.

ID del escenario	Escenario	Cantidad	Respuesta del sistema	Resultado de la prueba
EC 2.1	Realizar ejercicios “Al azar” exitosamente	V	Permite comenzar la ejercitación.	Permite comenzar la ejercitación.
EC 2.2	El número introducido en el campo “cantidad” es mayor	I	Señala el campo cantidad como incorrecto y muestra un <i>tooltip</i> con el	Señala el campo cantidad y muestra

Capítulo 3: Pruebas

	que la cantidad de ejercicios disponibles.		siguiente mensaje “Este valor está fuera del intervalo”.	un <i>tooltip</i> con el mensaje.
EC 2.3	El campo cantidad está vacío.	I	Señala el campo cantidad como incorrecto y muestra un <i>tooltip</i> con el siguiente mensaje “Este valor está fuera del intervalo”.	Señala el campo cantidad y muestra un <i>tooltip</i> con el mensaje.
EC 2.6	La cantidad de ejercicios seleccionada no es suficiente para la cantidad de estudiantes que iniciaron sesión.	I	Muestra el siguiente mensaje: “La cantidad de ejercicios seleccionada no es suficiente para todos los estudiantes autenticados en la sesión”.	Muestra el mensaje.

Tabla 3.4 Matriz de datos caso de prueba del CU Realizar ejercicios.

SC #3: Realizar ejercicios “Asignados”.

ID del escenario	Escenario	Disponibles	Asignados	Respuesta del sistema	Resultado de la prueba
EC 3.1	Realizar ejercicios “Asignados” exitosamente	N/A	V	Permite comenzar la ejercitación.	Permite comenzar la ejercitación.
EC 3.4	La cantidad de ejercicios seleccionada es menor que la cantidad de estudiantes.	N/A	I	Muestra el mensaje “La cantidad de ejercicios seleccionada no es suficiente para todos los estudiantes”.	Muestra el mensaje.
EC 3.15	Selecciona pasar ejercicios del área de seleccionados para el área de disponibles.	N/A	N/A	Elimina los ejercicios seleccionados del área de asignados y los muestra en el área de disponibles.	Elimina los ejercicios del área de asignados y los muestra en el

					área de disponibles.
--	--	--	--	--	----------------------

Tabla 3.5 Matriz de datos caso de prueba del CU Realizar ejercicios.

3.2 Resultados de las pruebas

Se planificaron 2 iteraciones en las cuales se probaron los casos de uso según el diseño de los casos de prueba. Se obtuvieron 13 no conformidades, 10 en la primera iteración y 3 en la segunda, las no conformidades detectadas fueron de funcionalidad, ortografía y correspondencia con documentación, las cuales fueron solucionadas. A continuación se muestra una gráfica de barras que ilustra la cantidad de no conformidades encontradas por tipo en cada iteración, seguidamente las tablas que las recogen en detalle.

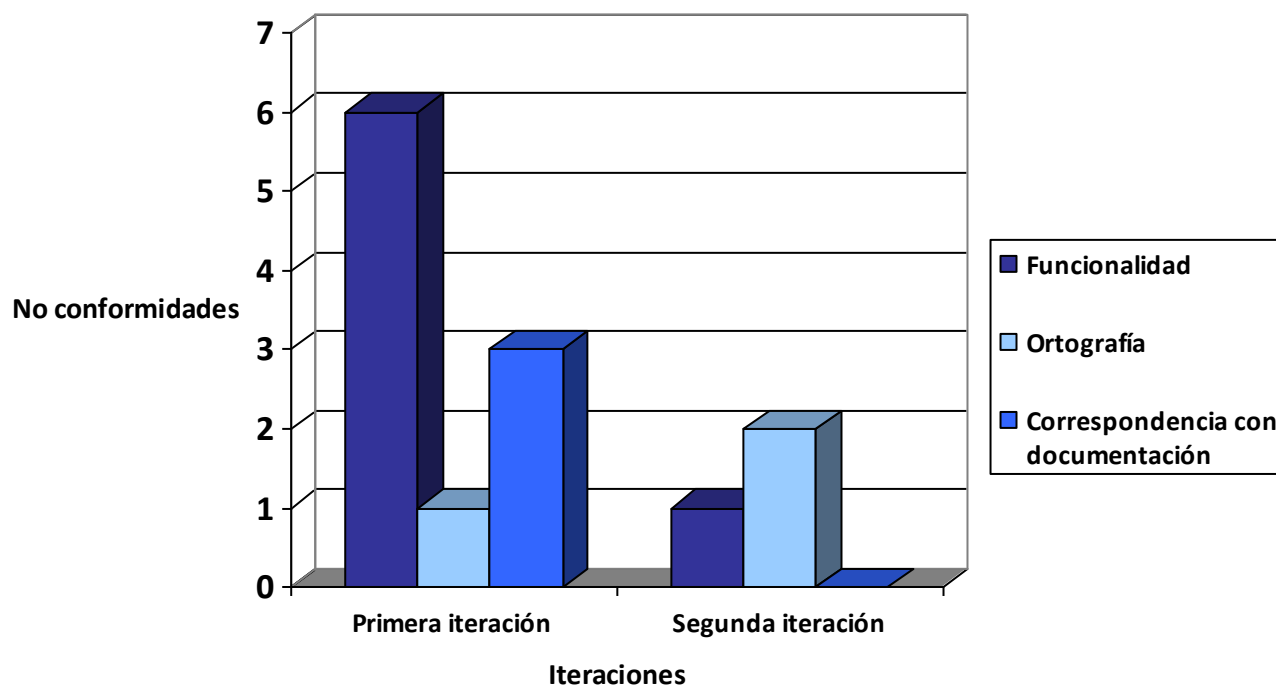


Figura 1: No conformidades por iteraciones

Figura

No conformidades del CU Realizar ejercicios

Elemento	No.	No conformidad	Aspecto correspondiente	Etapas de detección	Significativa
Revisa tu respuesta	1	Muestra un mensaje incorrecto.	La respuesta dada a un ejercicio es incorrecta y el mensaje mostrado por la aplicación informa que la respuesta es correcta.	1ra iteración.	NC de aplicación: Funcionalidad
Revisa tu respuesta	2	No muestra el mensaje afectivo de un ejercicio.	No muestra el mensaje afectivo para alentar al usuario en la resolución de un ejercicio.	1ra iteración.	NC de aplicación: Funcionalidad
Ejercicio siguiente	3	No muestra la respuesta de un ejercicio realizado.	Al seleccionar la opción anterior el ejercicio correspondiente ya había sido resuelto y no muestra la respuesta dada.	1ra iteración.	NC de aplicación: Funcionalidad
Revisa tu respuesta	4	La palabra prepárate está mal escrita.	En un mensaje conclusivo la palabra prepárate está escrita sin tilde.	1ra iteración.	NC de aplicación: Ortografía
Revisa tu respuesta	5	La palabra esfuérgate está mal escrita.	En un mensaje conclusivo la palabra esfuérgate está escrita sin tilde.	2da iteración.	NC de aplicación: Ortografía

Tabla 3.6 No conformidades del caso de uso Realizar ejercicios

No conformidades del CU Consultar resultados

Elemento	No.	No conformidad	Aspecto correspondiente	Etapas de detección	Significativa
Área ejercicios	1	La palabra evaluación se encuentra escrita sin tilde.	La palabra evaluación se encuentra escrita sin tilde.	2da iteración.	NC de aplicación: Ortografía

Tabla 3.7 No conformidades del caso de uso Consultar resultados

No conformidades del CU Gestionar ejercicio

Elemento	No.	No conformidad	Aspecto correspondiente	Etapas de detección	Significativa
Ver ejercicio	1	No se muestra la complejidad del ejercicio.	No se muestra la complejidad del ejercicio.	1ra iteración.	NC de aplicación: Correspondencia con documentación.
Eliminar ejercicio	2	El mensaje mostrado en la aplicación no concuerda con el mensaje de la ECU	Cuando se selecciona la opción eliminar el mensaje mostrado es "Está seguro que desea eliminar el ejercicio".	1ra iteración.	NC de aplicación: Correspondencia con documentación.

Tabla 3.8 No conformidades del caso de uso Gestionar ejercicio

No conformidades del CU Adicionar ejercicio

Elemento	No.	No conformidad	Aspecto correspondiente	Etapas de detección	Significativa
Adicionar Ejercicio de completar por escritura	1	No elimina el campo de una palabra asociada.	En la ventana que permite asociar palabras no elimina el campo de una palabra asociada.	1ra iteración.	NC de aplicación: Funcionalidad
Adicionar ejercicio de ordenar textos	2	El mensaje mostrado en la aplicación no concuerda con el mensaje de la ECU.	Al seleccionar la opción que permite desmarcar fragmentos muestra el mensaje "Serán desmarcadas XXX ¿Deseas continuar?"	1ra iteración.	NC de aplicación: Correspondencia con documentación.
Adicionar ejercicio de enlazar columnas	3	No elimina los elementos de las columnas.	Al seleccionar la opción que permite eliminar un elemento de las columnas no se eliminan.	1ra iteración.	NC de aplicación: Funcionalidad

Adicionar ejercicio de ordenar textos	4	El fragmento seleccionado para desmarcar se mantiene con el color de fondo.	Cuando se selecciona la opción que permite desmarcar fragmentos muestra el mismo con el color de fondo que tenía, o sea, no se desmarcó.	1ra iteración.	NC de aplicación: Funcionalidad
Adicionar ejercicio de seleccionar textos	5	El fragmento seleccionado para desmarcar se mantiene con el color de fondo.	Al seleccionar la opción que permite desmarcar fragmentos muestra el mismo con el color de fondo que tenía, o sea, no se desmarcó.	2da iteración.	NC de aplicación: Funcionalidad

Tabla 3.9 No conformidades del caso de uso Adicionar ejercicio

3.3 Conclusiones

En el presente capítulo se diseñaron los casos de prueba que guiaron la realización de las pruebas de caja negra, como resultado de la aplicación de las mismas se concluye que la cantidad de no conformidades detectadas no fue alarmante y con la erradicación de las mismas se logró obtener el módulo Ejercicios cumpliendo cabalmente con los requisitos especificados.

CONCLUSIONES GENERALES

Al término de esta investigación, se pudo arribar a las siguientes conclusiones:

- Se realizó un estudio sobre soluciones similares existentes, identificándose que ninguno de los consultados cumple con las características del módulo Ejercicios de la colección El Navegante en su versión multiplataforma. Además se estudiaron las tecnologías y herramientas utilizadas en el desarrollo de la solución propuesta.
- Se generaron los artefactos fundamentales que propone la metodología RUP para los flujos de trabajo Implementación y Prueba, entre los que se encuentran los Diagramas de componentes, que permitieron obtener un mejor entendimiento de la descomposición física del sistema en componentes, facilitando la implementación de los mismos; se diseñaron los casos de prueba que guiaron la realización de las pruebas de caja negra, las no conformidades detectadas fueron solucionadas, validando que el sistema responde a los objetivos planteados. Los artefactos obtenidos constituyen la documentación necesaria para garantizar el futuro mantenimiento del sistema.
- Se logró que la herramienta gestione de forma dinámica y fácil los contenidos educativos de los cuestionarios interactivos necesarios en el módulo Ejercicios, cumpliendo este con las especificaciones definidas para la realización de cuestionarios interactivos de diferentes tipologías, permitiendo además a equipos de trabajo ejercitar y comprobar sus conocimientos.

RECOMENDACIONES

- Se recomienda al proyecto Multisaber – Navegante mejorar el diseño de las interfaces de la herramienta de gestión de los cuestionarios interactivos, permitiendo que sean más amigables y agradables.
- Estandarizar el módulo Ejercicios con el objetivo de lograr una interoperabilidad de los cuestionarios interactivos gestionados, para que puedan ser reutilizados en plataformas educativas.

REFERENCIAS BIBLIOGRÁFICAS

1. **Verdecia Caballo, Enrique.** Edutec. Revista eletrónica de Tecnología Educativa. *Algunos Fundamentos filosóficos y Psicológicos de la Tecnología Educativa.* [En línea] [Citado el: 6 de diciembre de 2010.] <http://edutec.rediris.es/Revelec2/revelec23/everdecia/everdecia.html>.
2. **Valero Aguayo, Luis.** MÁQUINAS DE ENSEÑANZA DE SKINNER. [En línea] [Citado el: 6 de diciembre de 2010.] http://www.robertexto.com/archivo12/maquinas_ensienar.htm.
3. **Marquès Graells, Pere.** El software educativo. *Universidad Autónoma de Barcelona.* [En línea] [Citado el: 2 de diciembre de 2010.] http://www.lmi.ub.es/te/any96/marques_software/.
4. **González, Ysauro, Carmona Martínez, Victoria y Espíritu Reyes, Sara.** Evaluación de software educativo. *INSTITUTO LATINOAMERICANO DE LA COMUNICACIÓN EDUCATIVA.* [En línea] [Citado el: 2 de diciembre de 2010.] http://investigacion.ilce.edu.mx/panel_control/doc/c36,evaluacsoft.pdf.
5. **Galvis Panqueva, Álvaro.** [En línea] [Citado el: 2 de diciembre de 2010.] <http://www.slideshare.net/algavis50/ingeniera-de-software-educativo-1992-parte-1-fundamentos>.
6. **Gutiérrez Lam, José Antonio.** [En línea] [Citado el: 3 de diciembre de 2010.] http://www.ucp.pr.rimed.cu/sitios/revistamendive/nfuentes/num32/pdf/Art_1_Jose.pdf.
7. **Coloma Rodríguez, Orestes.** Portal Nacional del Fórum de Ciencia y Técnica. *HIPERENTORNO DE APRENDIZAJE "EUREKA": UN SOFTWARE EDUCATIVO PARA LA ENSEÑANZA DE LA MATEMÁTICA.* [En línea] [Citado el: 3 de diciembre de 2010.] www.forumcyt.cu/UserFiles/forum/Textos/1101687.pdf.
8. **Lamarca Lapuente, María Jesús.** Hipertexto: El nuevo concepto de documento en la cultura de la imagen. [En línea] [Citado el: 5 de diciembre de 2010.] <http://www.hipertexto.info/documentos/hipermedia.htm>.
9. Grupo de Diseño y Optimización de Sistemas Informáticos CES "Felipe II"– Aranjuez. Universidad Complutense de Madrid. *itest.* [En línea] [Citado el: 4 de diciembre de 2010.] dic 4.
10. **Noguera Rojas, Asdrúbal.** [En línea] [Citado el: 4 de diciembre de 2010.] <http://unimet.blog.com.es/>.
11. **González Verdejo, Noelia.** [En línea] [Citado el: 4 de dic de 2010.] http://www3.unileon.es/dp/dfh/noelia/met/trab2/trabajo2_sistautor.htm.
12. **Galli, Luca.** Luca Galli's Site. [En línea] [Citado el: 5 de diciembre de 2010.] <http://www.lucagalli.net/en/>.
13. Hot Potatoes. [En línea] [Citado el: 5 de diciembre de 2010.] <http://hotpot.uvic.ca/>.

14. webArdora.net. [En línea] [Citado el: 5 de diciembre de 2010.]
http://www.webardora.net/index_cas.htm.
15. Moodle. [En línea] [Citado el: 5 de diciembre de 2010.] <http://moodle.org>.
16. Entorno Virtual de Aprendizaje. [En línea] [Citado el: 5 de diciembre de 2010.]
<http://eva.uci.cu/mod/resource/view.php?id=200>.
17. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** *El proceso Unificado de Desarrollo de Software*. Madrid : Addison Wesley, 2000. 84-7829-036-2.
18. **G. Figueroa, Roberth, J. Solís, Camilo y A. Cabrera, Armando.** METODOLOGÍAS TRADICIONALES VS. METODOLOGÍAS ÁGILES. *Universidad Técnica Particular de Loja, Escuela de Ciencias en Computación*. [En línea] [Citado el: 10 de diciembre de 2010.]
<http://www.scribd.com/doc/42572078/Articulo-Metodologia-de-Sw-Formato>.
19. **Hernández Orallo, Enrique.** Universidad Politécnica de Valencia DISCA. [En línea] [Citado el: 8 de diciembre de 2010.] <http://www.disca.upv.es/enheror/pdf/ActaUML.PDF>.
20. Sitio de descargas de software. [En línea] [Citado el: 8 de diciembre de 2010.]
http://www.freedownloadmanager.org/es/downloads/Paradigma_Visual_para_UML_%28M%C3%8D%29_14720_p.
21. **Álvarez, Sara.** desarrolloweb.com. [En línea] [Citado el: 8 de diciembre de 2010.]
<http://www.desarrolloweb.com/articulos/sistemas-gestores-bases-datos.html>.
22. **Reyes Pupo, Hector Luis y Lavin Coello, Alejandro.** *Desarrollo de un módulo para gestionar los cuestionarios interactivos en la Colección El Navegante en su versión multiplataforma*. Ciudad de la Habana : s.n., 2010.
23. PostgreSQL. [En línea] [Citado el: 8 de diciembre de 2010.] <http://www.postgresql.org/about/>.
24. Definición de lenguaje de programación. [En línea] [Citado el: 10 de diciembre de 2010.]
<http://www.definicion.org/lenguaje-de-programacion>.
25. Lenguajes de Programación. [En línea] [Citado el: 7 de diciembre de 2010.] <http://www.lenguajes-de-programacion.com>.
26. Centro de Referencia en Accesibilidad y Estándares Web. *INTECO - Instituto Nacional de Tecnologías de la comunicación*. [En línea] [Citado el: 7 de diciembre de 2010.]
http://www.inteco.es/file/Kilat7eU_6hCnkU_DZrdhA.

Referencias bibliográficas

27. librosweb.es. [En línea] [Citado el: 7 de diciembre de 2010.]
<http://www.librosweb.es/javascript/capitulo1.html>.
28. PHP. [En línea] [Citado el: 7 de diciembre de 2010.] <http://www.php.net/manual/es/intro-whatism.php>.
29. **J. Gutiérrez, Javier.** *Departamento de Lenguajes y Sistemas Informáticos Universidad de Sevilla.* [En línea] [Citado el: 8 de diciembre de 2010.]
http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf.
30. librosweb.es. [En línea] [Citado el: 8 de diciembre de 2010.] http://www.librosweb.es/symfony_1_2/.
31. Eguiluz, Javier. *Maestros del Web.* [En línea] [Citado el: 8 de diciembre de 2010.]
<http://www.maestrosdelweb.com/editorial/el-framework-symfony-una-introduccion-practica-i-parte/>.
32. **Eguiluz, Javier.** *Maestros del Web.* [En línea] [Citado el: 8 de diciembre de 2010.]
<http://www.maestrosdelweb.com/editorial/javascript-facil-y-rapido-con-jquery/>.
33. babylon online dictionary. [En línea] [Citado el: 9 de diciembre de 2010.]
http://www.babylon.com/definition/entorno_de_desarrollo/Spanish.
34. NetBeans. [En línea] [Citado el: 9 de diciembre de 2010.] http://netbeans.org/index_es.html.
35. **Márquez Díaz, José, Sampedro, Leonardo y Vargas, Félix.** *Instalación y configuración de Apache, un servidor Web gratis.* [En línea] [Citado el: 9 de diciembre de 2010.]
http://ciruelo.uninorte.edu.co/pdf/ingenieria_desarrollo/12/instalacion_y_configuracion_de_apache.pdf.
36. **Myers, Glenford J.** *The art of software Testing.* s.l. : Segunda Edición. s.l. : Wiley, 2004.
37. **Juristo, Natalia, Moreno, Ana M. y Vegas, Sira.** *TÉCNICAS DE EVALUACIÓN DE SOFTWARE.* 2005.

BIBLIOGRAFÍA

1. **Marquès Graells, Pere.** El software educativo. Universidad Autónoma de Barcelona. [En línea] http://www.lmi.ub.es/te/any96/marques_software/.
2. **Galvis Panqueva, Álvaro.** [En línea] <http://www.slideshare.net/algalvis50/ingeniera-de-software-educativo-1992-parte-1-fundamentos>.
3. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** El proceso Unificado de Desarrollo de Software. Madrid : Addison Wesley, 2000. 84-7829-036-2.
4. **G. Figueroa, Roberth, J. Solís, Camilo y A. Cabrera, Armando.** METODOLOGÍAS TRADICIONALES VS. METODOLOGÍAS ÁGILES. Universidad Técnica Particular de Loja, Escuela de Ciencias en Computación. [En línea] <http://www.scribd.com/doc/42572078/Articulo-Metodologia-de-Sw-Formato>.
5. **Hot Potatoes.** [En línea] <http://hotpot.uvic.ca/>.
6. **Ardora.** [En línea] <http://www.webardora.net>.
7. Centro de Referencia en Accesibilidad y Estándares Web. INTECO - Instituto Nacional de Tecnologías de la comunicación. [En línea] http://www.inteco.es/file/Kilat7eU_6hCnkU_DZrdhA.
8. **Pressman, Roger S.** Ingeniería del Software. Un enfoque práctico. s.l.: McGraw-Hill Interamericana, 2005. 970-105473-3.
9. **PostgreSQL.** [En línea] <http://www.postgresql.org/about/>.
10. **LARMAN, Craig.** UML y Patrones. s.l.: Prentice Hall, 2000. 84-205-3438-2.
11. **Sommerville, Ian.** INGENIERÍA DEL SOFTWARE. Madrid: PEARSON EDUCACIÓN, 2005.84-7829-074-5.
12. **Reyes Pupo, Hector Luis y Lavin Coello, Alejandro.** Desarrollo de un módulo para gestionar los cuestionarios interactivos en la Colección El Navegante en su versión multiplataforma. Ciudad de la Habana : s.n., 2010.
13. **García Sánchez, Adrián y La Hoz Pavón, Lizandra.** Desarrollo del módulo Ejercicios para la plataforma educativa ZERA basado en estándares para la interoperabilidad. La Habana : s.n., 2011.

GLOSARIO DE TÉRMINOS

AJAX Asynchronous JavaScript And XML (JavaScript Asíncrono y XML): Es una técnica de desarrollo Web que permite crear aplicaciones interactivas. Estas aplicaciones se ejecutan desde el cliente, estableciéndose una comunicación asíncrona con el servidor, de forma que el contenido de las páginas se actualiza sin necesidad de volver a cargarlas. AJAX permite aumentar la interactividad, velocidad y usabilidad en las aplicaciones. No se establece como una única tecnología, sino como un conjunto de varias tecnologías que se pueden aplicar de formas muy variadas entre las que se incluyen XHTML y CSS, JavaScript, DOM, XML, XSLT y el objeto XMLHttpRequest.

XML: Extensible Markup Language (Lenguaje de marcado extensible), conjunto de reglas para definir etiquetas semánticas que organizan un documento en diferentes partes. Orientado principalmente al almacenamiento, procesamiento y transmisión de mensajes.

XSS: Del inglés Cross-site Scripting, es un tipo de inseguridad informática o agujero de seguridad basado en la explotación de vulnerabilidades del sistema de validación de HTML incrustado. Las vulnerabilidades de XSS ocurren cuando un atacante utiliza una aplicación web para mandar código malicioso, generalmente en la forma de un script, a un usuario diferente.

HTTP: HyperText Transfer Protocol. Protocolo de Transferencia de Hipertextos. Modo de comunicación para solicitar páginas Web.

CSRF: Aunque su nombre guarde cierta similitud con otro tipo de vulnerabilidades como Cross Site Scripting (XSS), hay importantes diferencias entre ellas. A diferencia de los ataques XSS, que se basan en explotar la confianza que tiene un usuario en un determinado sitio web o aplicación, los ataques CSRF, también denominados como Cross Site Reference Forgery o XSRF, se basan en explotar la confianza que los sitios web tienen con sus usuarios.

BSD: Berkeley Software Distribution (en español, Distribución de Software Berkeley). Pertenece al grupo de licencias de software libre. Esta licencia tiene menos restricciones en comparación con otras como la

GPL, estando muy cercana al dominio público. La licencia BSD, al contrario de la GPL, permite el uso del código fuente en software no libre.

Licencia GNU: General Public License (Licencia Pública General de GNU) es una licencia creada por la Free Software Foundation, está orientada principalmente a proteger la libre distribución, modificación y uso de software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

Retroalimentación: Es un proceso mediante el cual el sistema y el usuario interactúan por medio de mensajes, permiten al sistema expresar de forma afectiva la evaluación de la respuesta de un ejercicio, brindar una pista al usuario para que el mismo reflexione y responda de forma correcta; también mostrar información acerca de la respuesta correcta del ejercicio.

Multimedia: Es un término es muy utilizado en el contexto de las tecnologías de la información, en el ámbito digital se refiere a la integración en un sistema informático de textos, gráficos, imágenes, videos, animaciones, sonidos y cualquier otro medio que pueda ser tratado digitalmente.

Hipertexto: Es la organización de la información en diversos bloques de contenidos o nodos, conectados mediante enlaces en forma de red y permitiendo que los usuarios, según sus preferencias, puedan acceder a la información de forma no secuencial, es decir, no existe un orden único para acceder a cada nodo.

Multiplataforma: Programa o aplicación que puede utilizarse sin inconvenientes en distintas plataformas de hardware y sistemas operativos.

Navegador Web: Un navegador Web (del inglés, web browser) es un programa que permite visualizar la información que contiene una página Web, estando alojada en un servidor dentro de la World Wide Web o en uno local.