



Facultad 4

Desarrollo del módulo Juegos de la colección El Navegante en su versión multiplataforma

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

Autores: Alberto Hondal Hernández

Edilberto de la Peña Sarracén

Tutores: Ing. Nersa Doraines Acosta Labrada

Ing. Héctor Luis Reyes Pupo

La Habana, junio 2012

“Año 54 de La Revolución Cubana”

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores del trabajo titulado: “Desarrollo del módulo Juego de la colección El Navegante en la versión multiplataforma” y autorizamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de Junio del año 2010.

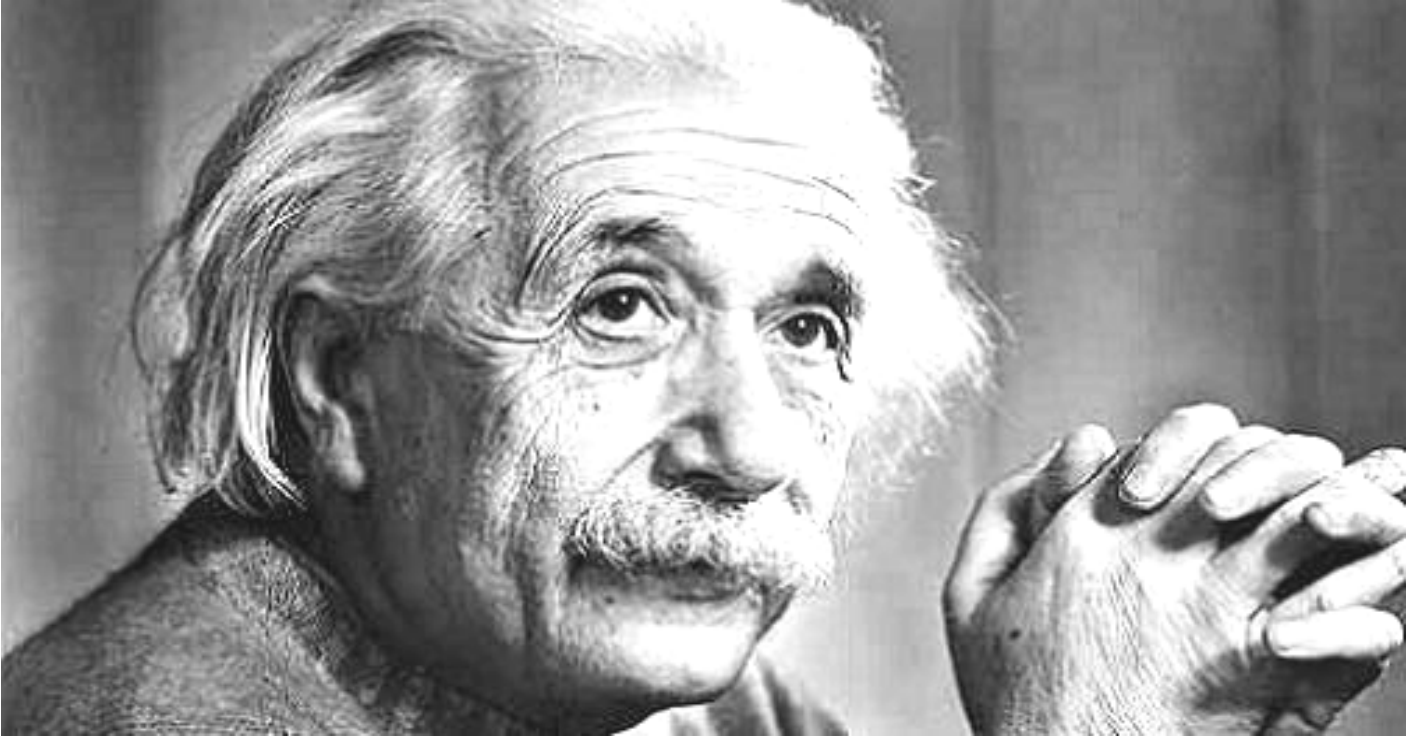
Firma del Autor
Alberto Hondal Hernández

Firma del Autor
Edilberto de la Peña Sarracén

Firma del Tutor
Ing. Héctor Luis Reyes Pupo

Firma del Tutor
Ing. Nersa Doraines Acosta Labrada

Pensamientos



“Los que dicen que algo es imposible no deberían molestar a los que lo están haciendo.”

“Hay una fuerza motriz más poderosa que el vapor, la electricidad y la energía atómica: la voluntad.”

Albert Einstein.

Agradecimientos

De Edilberto

A Dios por ser mi fortaleza, hoy puedo decir que todo se lo debo a Él.

A mi papá por apoyarme siempre en todo, y ser para mí un ejemplo a seguir, porque gracias a él hoy soy lo que soy, no me alcanzan las palabras para agradecerle a mi padre.

A mi mamá por ser la mejor madre del mundo, por estar siempre conmigo en todo momento, por darme la educación adecuada.

A mis padres es preciso agradecerles a los dos juntos, con todo el corazón infinitas gracias, podrían existir padres iguales pero mejores no los hay en el mundo, si hoy he llegado aquí es porque nunca soltaron mi mano, gracias por ser las personas más importantes de mi vida, nunca podré pagarles todo lo que les debo.

A mi hermana por ser mi inspiración y mi orgullo, por tener esa fe en mí, por todo lo que me has dado, gracias por existir, eres imprescindible en mi vida.

A toda mi familia que me vieron crecer y me ayudaron en todo, gracias por su aporte para que yo llegara aquí.

A mis compañeros de cuarto por el apoyo y por la unidad con que convivimos, a todos con los que compartí los años de carrera donde pasamos grandes momentos juntos.

Agradecimientos

De Alberto

A Dios por haber estado conmigo en todos los momentos difíciles en los cuales necesitaba de Él.

A mi papá por haber sido siempre exigente conmigo y demostrarme a mi mismo de que si puedo hacer las cosas cuando me las propongo, siempre te agradeceré todo lo que has hecho por mí a través de estos 5 largos años en la escuela, nunca podre recompensarte con nada, siempre te llevaré presente.

A mi mamá que ha sido mi ejemplo de sacrificio, de verdad mamá que te debo demasiado, desde que diste a luz siempre te he estado pidiendo todo creo que sin ti no hubiese podido terminar esta carrera, de verdad que te necesito a mi lado para todo.

A mis padres por haberme dado todo lo que necesitaba para llegar a ser lo que quiero, han sido mis guías desde el primer día que puse un pie en esta universidad, no creo que haya podido tener mejores personas en las que fijarme que en ustedes dos, les debo premiar con este título que más que yo ustedes se lo merecen, siempre estarán presentes en mi mente y mi corazón, los quiere su hijo mayor.

A mis hermanos que siempre me han seguido a todas partes que voy, les agradezco que siempre haya podido contar con ustedes para lo que sea.

Resumen

Las tecnologías de la información y las comunicaciones (TIC) han alcanzado gran auge a nivel mundial. Como parte de las transformaciones que se llevan a cabo en la sociedad cubana, han sido introducidas en el sistema educativo cubano las TIC. En Cuba se han creado tres importantes colecciones de software educativos: colección Multisaber para la enseñanza primaria, colección El Navegante distribuida en la enseñanza secundaria y colección Futuro para la educación superior. Debido a que Cuba se encuentra en proceso de migración al software libre, y que la colección El Navegante fue realizada con herramientas propietarias y solo se ejecuta en el sistema operativo Windows, surgió la necesidad de desarrollar la versión multiplataforma de la colección El Navegante, dirigida a los estudiantes de secundaria básica, la cual está compuesta por 6 módulos: Contenido, Ejercicios, Mediateca, Juegos, Maestro y Resultados.

El presente trabajo tiene como objetivo desarrollar el módulo Juegos de la colección El Navegante en su versión multiplataforma. Este módulo agrupa un conjunto de juegos que los estudiantes tienen a su disposición para ejercitar el contenido aprendido. El módulo está compuesto por: Descubre la imagen, Crucigrama, Sopa de letras, Ludo y Rompecabezas. Para guiar el desarrollo del módulo se utilizó Proceso Unificado de Rational (RUP) como metodología de desarrollo de software, PHP 5.3, JavaScript como lenguajes de programación, y Symfony, jQuery como framework de desarrollo del lado del servidor y del cliente respectivamente, además se utilizó UML como lenguaje de modelado y la herramienta Visual Paradigm como apoyo para realizar los diagramas y modelos pertinentes al sistema.

Palabras clave: software educativo, colección El Navegante, módulo Juegos.

Índice

Introducción	1
Capítulo 1: Fundamentación teórica.....	5
1.1 Introducción.....	5
1.2 Conceptos relacionados al software educativo	5
1.2.1 Software educativo.....	5
1.2.2 Multimedia	5
1.2.3 Hipermedia	6
1.2.4 Hiperentorno de enseñanza–aprendizaje.....	7
1.3 Análisis de sistemas similares existentes	7
1.3.1 Colección Multisaber.....	7
1.3.2 Colección El Navegante.....	7
1.3.3 Hiperentorno de aprendizaje Eureka.....	8
1.3.4 JUAdEA.....	8
1.3.5 Gcompris.....	9
1.3.6 Söchiki Walekerü.....	9
1.3.7 Resultado del estudio de colecciones de software educativos.....	9
1.3.8 Herramientas para generar cuestionarios.....	¡Error! Marcador no definido.
1.3.9 Resultado del estudio de las herramientas para generar cuestionarios interactivos.....	10
1.4 Metodología de desarrollo de software	11
1.4.1 Proceso Unificado de Desarrollo (RUP)	11
1.4.1.1 Características específicas de RUP	12
(La figura 1.1 muestra las fases de RUP con los flujos de trabajo).....	13

1.5 El Lenguaje Unificado de Modelado	13
1.6 Herramienta CASE de modelado.....	14
1.6.1 Visual Paradigm 3.4.....	14
1.7 Lenguajes de desarrollo	14
1.7.1 Lenguajes de programación del lado del cliente.....	15
1.7.1.1 XHTML.....	15
1.7.1.2 CSS 3	15
1.7.1.3 JavaScript 1.4	15
1.7.2 Lenguajes de programación del lado del servidor	16
1.7.2.1 PHP 5.3	16
1.8 Framework	16
1.8.1 Framework del lado del cliente.....	16
1.8.1.1 jQuery 1.5	16
1.8.2 Framework del lado del servidor	17
1.8.2.1 Symfony 1.4.3.....	17
1.9 Patrón Modelo Vista Controlador.....	17
1.10 Entorno de desarrollo integrado.....	17
1.10.1 NetBeans 6.9	18
1.11 Servidor web	18
1.11.1 Apache 2.0.....	18
1.12 Sistema de gestión de bases de datos	19
1.12.1 PostgreSQL 8.4	19
1.13 Conclusiones.....	19

Capítulo 2: Características del sistema.....	18
2.1 Introducción.....	18
2.2 Modelo de dominio.....	18
2.2.1 Análisis de los conceptos del dominio.....	18
2.2.2 Diagrama del modelo de dominio.....	20
2.3 Descripción del sistema propuesto.....	21
2.4 Especificación de requisitos.....	22
2.4.1 Requisitos funcionales.....	22
2.5 Modelo de casos de uso del sistema.....	25
2.5.1 Actores del sistema.....	25
2.5.2 Diagrama de casos de uso del sistema.....	25
2.5.3 Descripción de los casos de uso del sistema.....	26
2.6 Conclusiones.....	48
Capítulo 3: Análisis y diseño.....	49
3.1 Introducción.....	49
3.2 Modelo de análisis.....	49
3.2.1 Diagramas de clases del análisis.....	49
3.3 Modelo de diseño.....	52
3.3.1 Patrones de diseño.....	52
3.3.2 Diagramas de clases del diseño.....	52
3.4 Conclusiones.....	56
Capítulo 4: Implementación y prueba.....	60
4.1 Estándar de codificación.....	60

4.2 Modelo de implementación.....	60
4.2.1 Diagrama de componentes.....	60
4.2.2 Diagrama de despliegue.....	61
4.3 Modelos de datos.....	63
4.4 Prueba.....	65
4.5 Conclusiones.....	66
Conclusiones generales.....	67
Recomendaciones.....	68
Referencias bibliográficas.....	73
Glosario de términos.....	¡Error! Marcador no definido.
Anexos.....	¡Error! Marcador no definido.
Anexo 1: Descripción textual de los casos de uso.....	¡Error! Marcador no definido.
Anexo 2: Diagramas de clases del análisis.....	¡Error! Marcador no definido.
Anexo 3: Diagramas de clases del diseño.....	¡Error! Marcador no definido.
Anexo 4: Subsistemas de clases javaScript.....	¡Error! Marcador no definido.
Anexo 5: Estándares de codificación.....	¡Error! Marcador no definido.
Anexo 6: Diseño de los casos de pruebas.....	¡Error! Marcador no definido.

Introducción

Las Tecnologías de la Información y las Comunicaciones (TIC) se están desarrollando hoy día muy rápidamente y marcan notablemente la sociedad. Los profundos cambios que trae consigo este acelerado desarrollo se manifiestan en la actividad diaria del hombre, mejorando la calidad de vida de las personas y garantizando mayor cultura y beneficios económicos. Las TIC han venido transformando los procesos educativos, los que han sufrido diversas modificaciones en las formas de difundir y generar el conocimiento, además se evidencian cambios en los modos de intervención docente y en el desarrollo de nuevos espacios didácticos.

La dirección del gobierno cubano no está ajena a esta era de la información y el conocimiento. Desde el mismo triunfo de la Revolución se planteó un camino de desarrollo específicamente en la esfera de la educación, donde se trazó la meta de vincular las TIC a la educación en todos los niveles de enseñanza. Con la introducción de las TIC en este sector ha tomado protagonismo el software educativo, el cual es una herramienta de gran utilidad tanto para los alumnos como para los profesores, y se ha introducido en las escuelas con el fin de elevar la calidad del proceso de enseñanza-aprendizaje. Actualmente las escuelas cubanas disponen de colecciones de software educativo que apoyan este proceso, como es el caso de Multisaber para la enseñanza primaria, El Navegante para la enseñanza secundaria y Futuro para la enseñanza preuniversitaria.

La Universidad de las Ciencias Informáticas (UCI) trabaja sobre la base de la implementación de software educativos de alta calidad, estos software constituyen hiperentornos educativos o de aprendizaje. Con el desarrollo de estos programas informáticos la UCI se ha convertido en una importante empresa productora de software en Cuba.

La introducción de la colección El Navegante ha sido de gran utilidad en las escuelas cubanas pues los profesores pueden hacer llegar el conocimiento a los estudiantes de una forma más entretenida. Esta constituye un hiperentorno de enseñanza-aprendizaje realizado por el Ministerio de Educación (MINED) de Cuba. Actualmente está destinada a los tres niveles de la enseñanza secundaria y cuenta con un paquete de diez software educativos, cada uno de ellos compuesto por seis módulos básicos y diversos servicios informáticos. La colección fue realizada con herramientas propietarias y su uso es restringido solo para usuarios del sistema operativo Windows. Esta variante impide la utilización de sistemas operativos como Linux, Mac y otros.

El software libre se ha situado en lugares importantes en el mundo de la informática. Sus ventajas son principalmente que permite la libertad de uso y redistribución, además, brinda sistemas más seguros y económicos, al no tener que hacer frente al pago de grandes cantidades de dinero en licencias. En Cuba se está llevando a cabo el proceso de migración a software libre, y el sector de la educación no ha estado exento a estos cambios.

La colección El Navegante no es capaz de ajustarse a las nuevas tendencias de migración en las que se encuentra envuelto el (MINED) por la incompatibilidad que presenta el uso de esta en plataformas libres. Uno de los módulos de la colección es: Juegos, que cuenta con diversos juegos interactivos con los cuales los estudiantes recrean y demuestran las habilidades y conocimientos adquiridos, algunos constan de sistemas de preguntas y respuestas y otros se basan en la agilidad mental, así los estudiantes aprenden de una manera diferente, la principal limitación que presenta dicho módulo es que su uso es exclusivo para el sistema operativo Windows.

De lo planteado anteriormente se desprende el siguiente **problema a resolver**: ¿Cómo lograr que el módulo Juegos logre afrontar la migración a software libre en las escuelas cubanas?

Se precisó como **objeto de estudio** el software educativo con tecnologías multimedias.

Se definió como **objetivo general**: Desarrollar el módulo Juegos de la colección El Navegante en su versión multiplataforma.

El **campo de acción** de la investigación está enmarcado en el módulo Juegos de la colección El Navegante en su versión multiplataforma.

Objetivos específicos:

- Realizar el estudio del estado del arte.
- Delimitar las necesidades del cliente para un posterior análisis.
- Analizar y diseñar el módulo Juegos de la colección El Navegante.
- Implementar el módulo Juegos de la colección El Navegante.
- Probar el módulo Juegos de la colección El Navegante.

Tareas a cumplir:

- Análisis de la versión anterior del módulo Juegos de la colección El Navegante para la identificación de los requisitos generales.
- Entrevistas a los encargados de definir los requisitos de los juegos de la colección El Navegante.
- Estudio de la metodología a utilizar.
- Investigación de la tecnología a utilizar.
- Realización de los diagramas de clases del diseño del módulo Juegos de la colección El Navegante.
- Realización de los diagramas de implementación del módulo Juegos de la colección El Navegante.
- Implementación de los componentes del módulo Juegos de la colección El Navegante.
- Incorporación del módulo dentro de la línea base del producto general.
- Realización de las pruebas al módulo Juegos de la colección El Navegante.

Para el desarrollo de la presente investigación se emplearon los métodos teóricos:

- **Método de análisis histórico-lógico:** Este método se utilizará para estudiar y analizar los módulos Juego existentes. Además para el estudio de trabajos que traten sobre juegos educativos.
- **Método de modelación:** Este método se utilizará para la modelación de los diversos diagramas necesarios.
- **Método Analítico-Sintético:** Este método se utilizará para profundizar en los conceptos que se tratarán permitiendo un mejor entendimiento de lo que se quiere desarrollar.

Para la elaboración del presente trabajo de diploma se decidió dividir la información en 3 capítulos, los cuales quedan de la siguiente manera:

Capítulo 1: Fundamentación teórica: En este capítulo se realiza un estudio de los elementos teóricos que sustentan todo el trabajo. Se analizan los sistemas similares existentes y se estudian los lenguajes, la metodología, y herramientas a utilizar.

Capítulo 2: Características del sistema: En este capítulo se detallan los conceptos del sistema, se describe el modelo de dominio. Se identifican los requerimientos funcionales y los no funcionales, se elabora el diagrama de casos de usos y la descripción textual de cada uno de ellos.

Capítulo 3: Análisis y diseño: Se describe la solución a partir de los diferentes modelos y diagramas presentes en el flujo de trabajo de Análisis y diseño. Se describe la solución que se propone a partir de los diagramas de clases del análisis asociado a las funcionalidades del sistema y se representan los diagramas de clases del diseño, que reflejan una vista interna del sistema.

Capítulo 4: Implementación y Prueba: Se expone cómo está implementado el sistema en términos de componentes, a través de los diferentes diagramas, como son: el diagrama de componentes y de despliegue, obteniéndose una descripción de la implementación del sistema. Se describen los casos de pruebas por los cuales se prueba el producto.

Capítulo 1: Fundamentación teórica

1.1 Introducción

En este capítulo se hace un estudio detallado de los elementos fundamentales que componen la base teórica de la investigación. Se realiza una descripción de la metodología, herramientas y tecnologías a utilizar para dar solución a las tareas propuestas. Se abordan conceptos de gran importancia como software educativo, multimedia e hipermedia.

1.2 Conceptos relacionados al software educativo

Un software educativo es un programa didáctico que permite hacer llegar el conocimiento a una audiencia de manera más sencilla, además adapta el ritmo de trabajo y las necesidades de cada educando para lograr diversas alternativas de aprendizaje. (1)

1.2.1 Software educativo

Los software educativos interactúan con diferentes materias de diferentes maneras y ofrecen un entorno de trabajo más o menos sensible en dependencia de las circunstancias de los alumnos. (2) Se caracterizan por ser interactivos, a partir de que se le añaden programas de entretenimientos y recursos multimedia: videos, sonidos, fotografías, ejercicios, diaporamas y juegos instructivos, que apoyan la evaluación y el diagnóstico del estudiante. (3)

Aspectos comunes que distinguen al software educativo: (4)

- Son materiales desarrollados con finalidad didáctica.
- Utilizan la computadora como medio para realizar sus actividades.
- Son interactivos, ya que dan respuestas inmediatas a las acciones de los estudiantes.
- Permiten el intercambio de información entre la computadora y el estudiante.
- Se adaptan al ritmo de trabajo de cada estudiante, mostrándole ejercicios según las habilidades demostradas.
- Son fáciles de usar, no se requiere conocimientos avanzados en informática para su uso.

1.2.2 Multimedia

La multimedia es la combinación de las tecnologías que tiene el hombre: imágenes, sonidos, animación o video, música o efectos sonoros, y el texto. La tecnología multimedia permite, mediante la computadora,

combinar información de diversa naturaleza, permitiendo la posibilidad de interactuar dinámicamente con estos recursos. De este modo, se aproxima cada vez más a los modos del pensamiento humano, en virtud de la posibilidad de asociar, elegir y acceder a la información almacenada en la computadora. (6)

La multimedia se utiliza en muchas áreas. Pueden incluir simulaciones, programas de aprendizaje, presentaciones, informaciones y juegos. Dentro de los materiales multimedia se encuentran los materiales con propósitos educativos los cuales son denominados software educativo con tecnologías multimedia. Estos software se han convertido en algo muy importante dentro del sistema educacional, ya que facilitan mucho el proceso de enseñanza-aprendizaje. A través de estos programas no solo se regula el proceso de enseñanza de los estudiantes, sino que también se condicionan los procesos de aprendizaje, permiten motivar el interés de los estudiantes, atraer la atención de estos y así centrarlos en lo que se quiere enseñar, además permite evaluar a los estudiantes inmediatamente en dependencia de sus respuestas, permitiendo a estos ver las respuestas que da el ordenador. (7)

1.2.3 Hipermedia

La tecnología hipermedia es la integración del hipertexto y la multimedia, esta es la organización de imágenes, videos, textos, sonidos y cualquier otro medio digitalizado en diversos nodos interconectados mediante enlaces. El hipertexto tiene los siguientes elementos: nodos y enlaces. Los nodos conforman las partes del hipertexto que contienen la información que es accedida por los usuarios y los enlaces o hiperenlaces permiten navegar entre los distintos nodos de una manera no lineal. La hipermedia facilita la interacción del usuario con la información de una manera amigable y sencilla. (8)

En un hipermedia se pueden utilizar dos formas básicas para consultar la información que se encuentra almacenada en sus nodos: la navegación y la búsqueda. La navegación es considerada la forma primaria de acceso a la información en un hipermedia y consiste en recorrer los nodos a través de los enlaces, consultando la información que en él se almacena. En la búsqueda el usuario solicita al sistema el acceso directo a un determinado dato. (9)

La tecnología hipermedia es muy utilizada para la producción de aplicaciones informáticas en muchas ramas de la sociedad, esta conforma la base tecnológica para la creación de software educativo en Cuba. Es muy utilizada con fines docentes en todo el mundo ya que provee una estructura de navegación a

través de una gran variedad de medios, puede tener componentes dinámicos y también puede ser controlada por el usuario.

1.2.4 Hiperentorno de enseñanza–aprendizaje

Los hiperentornos de enseñanza-aprendizaje consisten en la combinación de las diferentes tipologías de software educativos: libros electrónicos, sistemas tutoriales, simuladores, juegos educativos y sistemas de expertos, basados en tecnología hipermedia. Estos hiperentornos, que tienen en común la utilización de un ambiente informático con una intención didáctica, integran a los estudiantes y profesores con los recursos informáticos. Son técnicas que ayudan al sistema educacional a evaluar a los estudiantes mediante las tecnologías informáticas y brindan la posibilidad a los estudiantes de interactuar con los contenidos que le permiten el control de su aprendizaje. (10)

1.3 Análisis de sistemas similares existentes

En este epígrafe se tratarán los aspectos más importantes de las colecciones Multisaber y El Navegante existentes destinadas a las enseñanzas primaria y secundaria respectivamente. Además se hace un estudio de diferentes juegos educativos, así como las herramientas para generar cuestionarios a fin de obtener información sobre cómo se desarrollan y qué elementos fundamentales se tienen en cuenta en este tipo de aplicaciones.

1.3.1 Colección Multisaber.

La colección Multisaber es un grupo de multimedias educativas destinadas a la enseñanza primaria, está compuesta por treinta y dos software y tiene seis módulos: Temas, Ejercicios, Biblioteca, Profesor, Resultados y Juegos. Multisaber posee una presentación genérica y una particular de cada producto. El nombre Multisaber para la colección expresa la idea del aprendizaje de múltiples contenidos para los niños desde temprana edad. Los software de Multisaber poseen una interfaz estandarizada, que proporcionan un ambiente de trabajo amigable e intuitivo con alto nivel de interactividad para hacer más amena la enseñanza y más efectivo el aprendizaje en los estudiantes. Esta colección contiene un módulo Juegos similar al módulo que se va a desarrollar en la nueva versión de la colección El Navegante.

1.3.2 Colección El Navegante.

La colección El Navegante instalada en las escuelas secundarias de Cuba cuenta con diez productos, en cada uno están presentes 6 módulos: Contenido, Mediateca, Ejercicios, Juegos, Resultados y Maestro. El

módulo Juegos es el encargado de reunir los diversos juegos didácticos con los cuales los estudiantes demuestran lo asimilado de una manera divertida, a la vez que aprenden a través de los retos que se les plantean. Crucigramas, Ludo, Sopa de letras y Descubre la imagen son algunos de los juegos presentes en la colección, todos estos juegos son educativos pues combinan contenidos de Matemáticas, Literatura y Español, Inglés, Educación Laboral y Dibujo Básico, Historia Antigua y Medieval.

El Navegante permite el trabajo grupal para aprovechar el aprendizaje colaborativo, incorpora recursos multimedia como imágenes fijas y en movimiento, sonidos y videos. La colección se caracteriza por el empleo de estrategias pedagógicas de análisis de respuestas y retroalimentaciones reflexivas. Registra la actuación de los estudiantes con el software para su empleo en el análisis de los resultados obtenidos usados para la toma de decisiones en la dirección del proceso pedagógico. (11)

1.3.3 Hiperentorno de aprendizaje Eureka.

Como parte de la colección Futuro para el preuniversitario, se encuentra el software Eureka, destinado a apoyar la enseñanza de la Matemática en este nivel y está compuesto por siete módulos para dar apoyo a los diversos momentos del proceso docente-educativo. Permite la gestión de usuarios pero no del contenido. (38)

1.3.4 JUAdEA.

Software educativo titulado “Jugando se aprende de Educación Ambiental”. Su tema central es la problemática ambiental de la provincia de Cienfuegos. Este software está dirigido a estudiantes de noveno grado por tener conocimientos ambientales previos, aunque puede ser utilizado por otros grados de este nivel de enseñanza. Consta de 8 módulos: Principal, Introducción, Contenido, Biblioteca, Curiosidades, Juegos, Profesor y Créditos. El software fue implementado para entornos de Windows 32 bits y como principal herramienta en el diseño de sus imágenes fue utilizado el Software Adobe Photoshop versión 7.0. (39)

Como fase posterior a su diseño fue necesario el ensamblaje del software mediante la herramienta Sprite Kernel, elaborada por el Grupo de Producción de Software de la Universidad de Cienfuegos, todo esto bajo el lenguaje de programación Object Pascal y utilizando el compilador Borland Delphi 7. El diseño de los juegos incluye imágenes y se basa en ejercicios de tipo marcar y completar oraciones. El contenido no se puede editar, es obligatorio trabajar con el predefinido.

1.3.5 GCompris.

GCompris es un programa educativo especialmente diseñado para niños de entre 2 y 10 años. Es software libre y está disponible para Windows, Mac y Linux. Contiene una colección formada por más de 50 actividades y juegos dedicados a tratar todo tipo de temas:

- Uso del ordenador y sus periféricos: acostumbrarse al teclado, ratón.
- Álgebra: memoria, números, tablas, imágenes.
- Ciencia: el ciclo del agua, electricidad, canales, el submarino.
- Geografía: países y mapas.
- Juegos: ajedrez, memoria, fútbol, rompecabezas.
- Lectura: diferentes ejercicios de lectura.
- Otros: las horas, dibujos vectoriales.

Muchas actividades y juegos tienen varios niveles de dificultad, por lo que pueden ser usados por personas de casi cualquier edad. Los niños son usuarios muy especiales y exigentes, y la interfaz está muy trabajada para que sea lo más sencilla, agradable y cómoda posible. (40)

1.3.6 Süchiki Walekerü.

El software está constituido por cuatro módulos: escuchar el cuento, jugar con el cuento, ayuda y salida del sistema. En el módulo de juegos hay una serie de actividades que en su mayoría buscan incentivar en los niños y maestros el uso de la lectura y escritura del wayuunaiki (Idioma guajiro empleado en La Guajira, Colombia y en Zulia, Venezuela), tales actividades están conformadas por identificación de vocales, sopa de letras, ejercicios de completar, escribir palabras luego de escuchadas, asociar pares de palabras con imagen, ordenar oraciones para formar párrafos con sentido, juegos de memoria con palabras y rompecabezas, entre otros. Su objetivo es incentivar la lectura y escritura del wayuunaiki y está constituido por 12 tipos de juegos que en total conforman 23 juegos. El programa seleccionado para desarrollar el software fue el Flash MX de Macromedia. (41)

1.3.7 Hot Potatoes

Hot Potatoes es un conjunto de seis herramientas de autor desarrollado por el equipo del University of Victoria CALL Laboratory Research and Development, que permiten elaborar ejercicios interactivos basados en páginas Web de seis tipos básicos. La interactividad de los ejercicios se consigue mediante JavaScript. Entre sus principales componentes se encuentra **JCross**, este se encarga de crear

crucigramas, puede usarse una cuadrícula de cualquier tamaño. Un botón de ayuda permite al estudiante solicitar una letra en el caso de que la necesite.

1.3.8 EdiLIM

EdiLIM es un software para crear materiales educativos, especialmente ejercicios aplicables a la docencia y accesibles en forma web. Permite crear los siguientes tipos de páginas interactivas (43):

Puzzle: Resolver un rompecabezas con la posibilidad de ver la imagen con distintos niveles de transparencia. Las piezas se arrastran con el ratón y se ajustan automáticamente.

Sopa de letras: Para buscar palabras mezcladas entre un cuadro de letras de 10 por 10. Puede aparecer la descripción de las palabras a buscar, directamente las palabras o imágenes. Se debe pulsar sobre la primera letra de la palabra y mover el ratón hasta la última.

Arrastrar textos/arrastrar imágenes: Mover con el ratón las piezas hasta encajarlas correctamente.

Mover imágenes: Arrastrar hasta seis imágenes. Permite encajar figuras, completar imágenes, establecer secuencias, etc.

Completar: Llenar huecos mediante la escritura de la respuesta o moviendo cuadros de texto.

Palabra secreta: Descubrir la palabra oculta pulsando sobre las letras, hasta seis posibilidades de error. Admite imágenes.

1.3.9 Resultado del estudio de los sistemas similares.

Al estudiar las características de las colecciones descritas, se pueden resaltar elementos positivos como la utilización de perfiles que garantizan el acceso de los estudiantes solamente a la información predefinida por el profesor, brindando nuevas oportunidades de trabajo diferenciado según las necesidades de cada perfil. Un aspecto importante también es la utilización de niveles de dificultad para controlar la complejidad del juego. Una característica que arroja la investigación es que en el desarrollo de los juegos se ha optado por utilizar software propietario o por centrarse solo en la compatibilidad con el sistema operativo Microsoft Windows. Además una restricción que presentan estas colecciones es que no permiten la gestión del contenido.

Si bien las herramientas mencionadas son capaces de generar muchos tipos de juegos, presentan grandes limitaciones, no solo en cuanto a la necesidad de adquirir licencias, sino también en la capacidad de adaptarse a los contenidos específicos de algunos de los juegos a incluir en la colección El Navegante en su versión multiplataforma.

1.4 Metodología de desarrollo de software

Una metodología de desarrollo de software es un conjunto de pasos y procedimientos que deben seguirse para desarrollar un software con calidad. Define las etapas para el desarrollo de un software, las tareas que se deben llevar a cabo en cada etapa, las restricciones que deben aplicarse, las técnicas y herramientas que deben emplearse, y la forma de controlar y gestionar un proyecto.(12)

Las metodologías existentes se pueden clasificar en dos grupos: tradicionales y 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 Proceso Unificado de Rational (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 son adaptables 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. (12)

En el momento que un equipo decide realizar una aplicación, se debe hacer un estudio minucioso de las metodologías existentes, para escoger la más adecuada a utilizar teniendo en cuenta las características, complejidad y envergadura del proyecto.

1.4.1 Proceso Unificado de Desarrollo (RUP)

La metodología que se utilizará para guiar el proceso de desarrollo del módulo Juegos de la colección El Navegante es RUP, esta es la metodología que está definida para todos los módulos de dicha colección, además el equipo de desarrollo no tiene una comunicación directa e inmediata con los clientes, toda la interacción es a través de terceros, también permite obtener una completa documentación del proceso de desarrollo. Esta metodología captura las mejores prácticas del conocimiento de líderes en ingeniería de software y proporciona a los equipos de desarrollo guías, estándares y recomendaciones para la construcción de software de alta calidad. (13)

1.4.1.1 Características específicas de RUP (13)

Dirigido por casos de uso: los casos de uso representan los requisitos funcionales del sistema. En RUP los casos de uso además de ser una herramienta para especificar los requisitos del sistema, también se utilizan para guiar el análisis, diseño, implementación y prueba del proceso de desarrollo del software. Los casos de uso se especifican y diseñan al principio de cada iteración y son la fuente a partir de la cual los ingenieros de prueba construyen sus casos de prueba.

Centrado en la arquitectura: la arquitectura involucra los elementos más significativos del sistema y está influenciada, entre otros, por las plataformas de software, sistemas operativos, sistemas de gestión de bases de datos, además de otros como sistemas heredados y requerimientos no funcionales. Es el eje central para el desarrollo de un software, esta sirve para que todos los implicados en el desarrollo tengan una idea clara de qué es lo que se está construyendo.

Iterativo e incremental: el proceso de desarrollo del software se puede dividir en varias iteraciones, cada una debe ser considerada como un mini proyecto cuyo núcleo fundamental está constituido por una o más iteraciones de las actividades principales básicas de cualquier proceso de desarrollo. Los incrementos del software se refieren al crecimiento del producto.

RUP está constituido por 9 flujos de trabajo (los 6 primeros son conocidos como flujos de ingeniería y los 3 últimos de apoyo): modelamiento del negocio, requisitos, análisis y diseño, implementación, prueba, instalación, administración de configuración y cambios, administración de proyectos y ambiente, los cuales tienen lugar sobre 4 etapas o fases: inicio, elaboración, construcción y transición. (14)

Fases de RUP (15)

Inicio: se describe el negocio y se delimita el proyecto, se identifican los casos de uso del sistema y los riesgos.

Elaboración: se define la arquitectura del sistema. Se realiza un plan de proyecto y se mitigan la mayoría de los riesgos.

Construcción: se logra obtener un producto listo para ser utilizado, el cual cuenta con un manual de usuario.

Transición: la versión final ya se encuentra lista para entregar e instalar al cliente, durante esta fase se pueden reparar algunos errores del producto.

(La figura 1.1 muestra las fases de RUP con los flujos de trabajo).

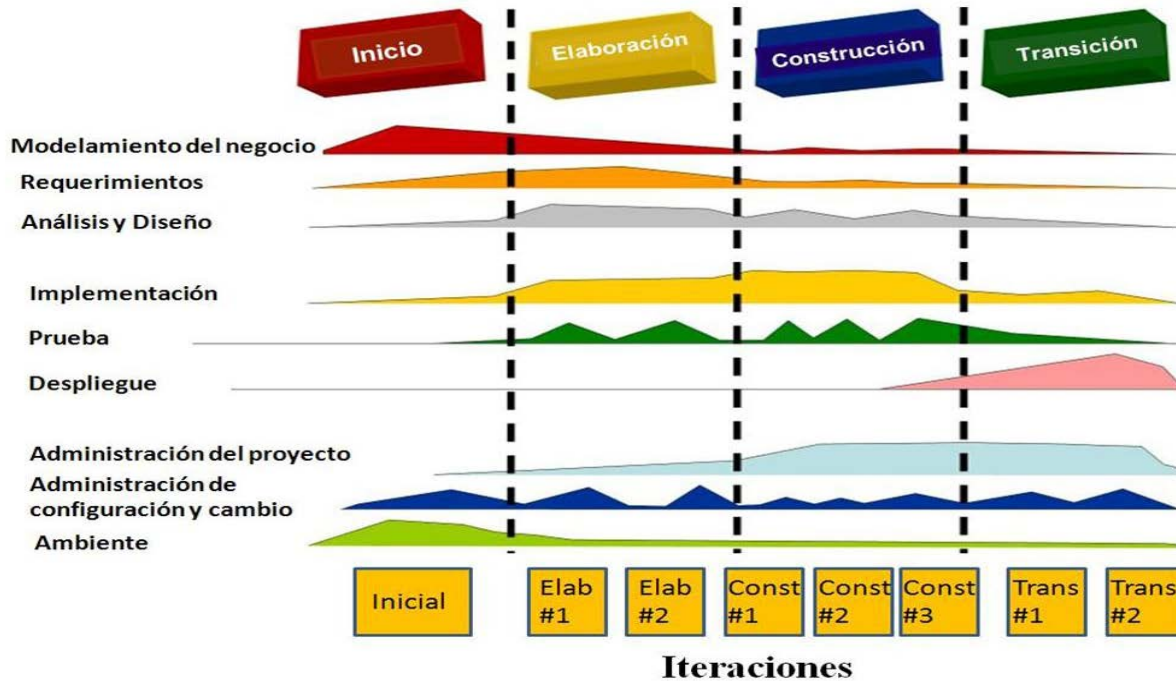


Figura 1.1 Fases y flujos de trabajo de RUP

1.5 El Lenguaje Unificado de Modelado

El Lenguaje Unificado de Modelado (UML) ofrece soporte para clases, clases abstractas, relaciones, comportamiento por iteración, empaquetamiento, entre otros. Estos elementos se pueden representar mediante nueve tipos de diagrama, que son: de clases, de objetos, de casos de uso, de secuencia, de colaboración, de estados, de actividades, de componentes y de desarrollo. (16)

UML permite representar el sistema de una forma gráfica, de esta manera se puede entender y especificar más fácilmente sus características. Es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema. Permite realizar una verificación y validación del modelo realizado. La arquitectura del proyecto definió que UML se utilizara como lenguaje de modelado.

1.6 Herramienta CASE de modelado

Las herramientas CASE (Computer Aided Software Engineering, o Ingeniería de Software Asistida por Computadora) son un conjunto de métodos, utilidades y técnicas que ayudan al desarrollo de software. Aumentan la productividad y logran una reducción en el costo de tiempo y de dinero. Al utilizar estas herramientas se puede abstraer al código fuente en un nivel donde la arquitectura y el diseño son más fáciles de entender y modificar. (17)

1.6.1 Visual Paradigm 3.4

Visual Paradigm 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. Ayuda a una construcción más rápida de aplicaciones, con un menor costo y mejor calidad. Permite realizar todos los tipos de diagramas de clases, obtener código inverso y generar código desde diagramas en una amplia gama de lenguajes, dentro de estos PHP. Otra de las características es que es multiplataforma, soporta todos los diagramas UML, diagramas de SysML y el diagrama entidad-relación. También produce documentación del sistema en formato PDF y HTML. Se utilizará Visual Paradigm como herramienta CASE, ya que la arquitectura del proyecto definió que se utilizara esta en el desarrollo del módulo Juegos de la colección El Navegante, la UCI tiene comprada la licencia de esta herramienta.

1.7 Lenguajes de desarrollo

Con el lenguaje de programación es que los programadores dan instrucciones a una computadora, está compuesto por un conjunto de reglas sintácticas y semánticas que son interpretadas por la máquina. Hoy en día existen diversos lenguajes de programación para desarrollar en la web que permiten interactuar con los usuarios y utilizan sistemas de base de datos, se debe tener muy claro las características y posibilidades del equipo de desarrollo y del proyecto a la hora de seleccionar el lenguaje de programación a utilizar. Los lenguajes de programación para las páginas web se clasifican en lenguajes del lado del cliente y lenguajes del lado del servidor; se utilizarán en el lado del cliente: XHTML, CSS y JavaScript, y en el lado del servidor, se utilizará: PHP. Todos estos lenguajes fueron seleccionados por el proyecto para la construcción de la colección El Navegante.

1.7.1 Lenguajes de programación del lado del cliente

1.7.1.1 XHTML

XHTML (Lenguaje de Marcado de Hipertexto Extensible) es una versión más estricta y limpia de HTML. Al estar orientado al uso de un etiquetado correcto, exige una serie de requisitos básicos a cumplir en lo que a código se refiere. Entre estos requisitos básicos se puede mencionar una estructuración coherente dentro del documento donde se incluirían elementos correctamente anidados, etiquetas en minúsculas, elementos cerrados correctamente y atributos de valores entrecomillados. XHTML aumenta la velocidad de desarrollo y de navegación, ya que permite crear y maquetar páginas webs correctamente, las cuales se carga más rápido en los navegadores. (18)

1.7.1.2 CSS 3

Cascading Style Sheets u Hojas de Estilo en Cascada es un lenguaje creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. El objetivo de CSS es separar la estructura de un documento de su presentación, siendo muy útil cuando se quiere cambiar un aspecto del diseño de un sitio web. El lenguaje CSS brinda la posibilidad de tener el control del diseño en la aplicación.

1.7.1.3 JavaScript 1.4

JavaScript es un lenguaje de script multiplataforma, no es necesario compilar los programas para ejecutarlos, los programas escritos con este lenguaje se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. Es pequeño y ligero, no es útil como un lenguaje independiente, más bien está diseñado para una fácil incrustación en otros productos y aplicaciones. 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. Es dinámico, responde a eventos en tiempo real. (19)

JavaScript permite tener el control total de las animaciones y la manipulación de los componentes en la página web. Es ideal la utilización de JavaScript en el desarrollo del módulo debido a la gran cantidad de animaciones que contiene este.

1.7.2 Lenguajes de programación del lado del servidor

1.7.2.1 PHP 5.3

PHP (acrónimo de PHP: Hypertext Preprocessor) es un lenguaje muy utilizado para la generación de sitios web dinámicos, es un lenguaje multiplataforma. Tiene capacidad de conexión con la mayoría de los manejadores de base de datos. Posee documentación en su página oficial, la cual incluye descripción y ejemplos de cada una de sus funciones (20). PHP brinda la posibilidad de programar orientado a objetos, además es un lenguaje libre y fácil de aprender.

1.8 Framework

El framework es una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación, promueve buenas prácticas de desarrollo como el uso de patrones. Permite la reutilización del código y facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas. (21)

Estas estructuras simplifican el desarrollo de una aplicación y agilizan el desarrollo de sistemas web. Un framework asiste al trabajo de un programador en dos niveles: en el lado del cliente y en el lado del servidor, en el lado del cliente se usará: jQuery y en el lado del servidor: Symfony.

1.8.1 Framework del lado del cliente

1.8.1.1 jQuery 1.5

jQuery es una biblioteca JavaScript muy rápida y ligera, incluye muchas utilidades para crear fácilmente las páginas web de aplicaciones dinámicas complejas. Es un producto serio, estable, bien documentado, con un gran equipo de desarrolladores a cargo de su mejora y actualización. Es una librería muy utilizada por su sencillez, flexibilidad y rendimiento. (22)

Esta biblioteca facilita la creación de aplicaciones complejas del lado del cliente ya que brinda ayuda en la creación de interfaces de usuario, para animar el contenido de las páginas de forma dinámica. Además de todo lo mencionado, este framework tiene gran aceptación por parte de los programadores, por lo que es muy útil para el desarrollo del módulo Juegos de la colección El Navegante.

1.8.2 Framework del lado del servidor

1.8.2.1 Symfony 1.4.3

Symfony es uno de los framework más documentado del mundo, cuenta con miles de páginas de documentación distribuidas en varios libros gratuitos y decenas de tutoriales. Aumenta exponencialmente la productividad y ayuda a mejorar la calidad de las aplicaciones web. (23)

Características de Symfony: (24)

- Fácil de instalar y configurar en la mayoría de las plataformas.
- Independiente del sistema gestor de bases de datos.
- Sencillo de usar en la mayoría de los casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.

Es un framework basado en el lenguaje de desarrollo PHP, posee integración con el sistema gestor de base de datos PostgreSQL. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja.

1.9 Patrón Modelo Vista Controlador

El patrón arquitectónico Modelo-Vista-Controlador (MVC) se utiliza mucho para el desarrollo de aplicaciones web, por la posibilidad que brinda de organizar el código de una aplicación separando los datos de la aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Define una separación entre datos y presentación, y presenta gran sencillez para crear distintas representaciones de los mismos datos. Disminuye el esfuerzo de los desarrolladores a la hora de programar, les permite la reutilización de los componentes y desarrollar prototipos rápidos. El modelo representa toda la información con que trabaja el sistema, la vista es la página con que interactúa el usuario, con esta se les muestra la información a estos, el controlador maneja la interacción y peticiones de los usuarios, y los cambios en el modelo o en la vista (25). Este patrón arquitectónico es la base de Symfony

1.10 Entorno de desarrollo integrado

Un entorno de desarrollo integrado (Integrated Development Environment, IDE) es un programa compuesto por un conjunto de herramientas muy útiles, las cuales son utilizadas por los programadores, tales como depuradores de código, un editor de texto, un compilador, un intérprete, entre otros. Puede ser utilizado con un único lenguaje de programación o con varios. (26)

1.10.1 NetBeans 6.9

NetBeans es un IDE de código abierto, gratuito, sin restricciones de uso, ofrece una amplia documentación y recursos de capacitación. Es una plataforma que permite a los desarrolladores la creación rápida de diferentes tipos de aplicaciones, esta herramienta sirve para escribir, compilar, depurar y ejecutar programas. (27)

Es muy importante para la implementación del módulo Juegos de la colección el Navegante el NetBeans, ya que éste presenta soporte a los frameworks Symfony y JQuery, y tiene soporte para lenguajes como PHP, JavaScript, HTML, CSS. Además se integra con los sistemas de control de versiones tales como SVN.

1.11 Servidor web

Un servidor web es un programa encargado de atender y responder diferentes tipos de peticiones realizadas por los navegadores web usando protocolos HTTP y HTTPS. Su funcionamiento se basa en estar a la espera de peticiones, después de recibir esta, busca el recurso solicitado y luego lo envía utilizando la misma conexión por la que recibió la petición.

1.11.1 Apache 2.0

El servidor Apache es un software que está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías:

- módulos base: con las funciones básicas.
- módulos multiproceso: son los responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atenderlas.
- módulos adicionales: es cualquier otro módulo que le añada una funcionalidad al servidor. (28)

Presenta las siguientes características: (29)

- Es gratuito y de código abierto.
- Posee una amplia documentación.
- Es robusto y seguro.
- Funciona en muchos sistemas operativos.
- Es un servidor altamente configurable de diseño modular.

Además de las características mencionadas el servidor Apache es multiplataforma y se caracteriza por ser un servidor web flexible, rápido y eficiente. Apache se integra con los lenguajes de programación PHP y JavaScript, con el sistema gestor de base de datos PostgreSQL y con el IDE NetBeans.

1.12 Sistema de gestión de bases de datos

Un sistema de gestión de base de datos es un software que permite introducir, organizar y recuperar la información de las bases de datos. Ayuda a realizar acciones como: definición, mantenimiento de la integridad, control de la seguridad, privacidad y manipulación de los datos. (30)

1.12.1 PostgreSQL 8.4

Como sistema gestor de base de datos se utilizará PostgreSQL, este es un gestor de base de datos relacional, libre, orientado a objetos. Funciona en todos los principales sistemas operativos, incluyendo Linux, UNIX (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64), y Windows. Además brinda una amplia documentación lo que hace más fácil trabajar con el mismo. (31)

PostgreSQL es compatible con el almacenamiento de objetos binarios, incluyendo imágenes, sonidos y videos. Tiene interfaces de programación con diferentes lenguajes como PHP, C, C++, Java, .NET, Perl, Python, entre otros. Es muy eficiente cuando se desea almacenar un gran número de datos.

1.13 Conclusiones

En este capítulo se estudiaron y analizaron los temas principales para el desarrollo del módulo Juegos de la colección El Navegante. Se realizó un estudio detallado sobre los conceptos relacionados al proceso del software educativo. Se realizó un análisis de los sistemas similares existentes, demostrándose la necesidad de dicho módulo. Se investigó sobre las tecnologías y herramientas a utilizar. Se utilizará RUP como metodología a emplear. Se utilizará como lenguaje de modelado UML, y como herramienta CASE de modelado Visual Paradigm 3.4. Los lenguajes de desarrollo que se utilizarán, como lenguajes de programación del lado del cliente: XHTML, CSS 3, JavaScript 1.4, y como lenguaje de programación del lado del servidor PHP 5.3. Se utilizarán JQuery 1.5 como frameworks del lado del cliente y Symfony 1.4.3 como frameworks del lado del servidor. Para la organización del módulo el encargado será el patrón arquitectónico Modelo Vista Controlador y el entorno de desarrollo integrado a utilizar será NetBeans 6.9 que brinda soporte a Symfony y JQuery. Como servidor Web, Apache 2.0 se usará por ser entre otras

razones un servidor web flexible, rápido y eficiente. El sistema gestor de base de datos será PostgreSQL 8.4.

Capítulo 2: Características del sistema

2.1 Introducción

En el presente capítulo se realiza la descripción de la propuesta de solución, se da una visión de las características del módulo a desarrollar. Se aclaran los conceptos más importantes del dominio con la confección del modelo de dominio con el objetivo de comprender el contexto del sistema. Se identifican los requerimientos funcionales y no funcionales, que son las características y cualidades que el sistema debe cumplir. Se obtienen y describen detalladamente los casos de uso que guían la solución del sistema y se realiza el diagrama de casos de uso del sistema para reflejar las relaciones entre los actores del sistema y las secuencias de acciones en las que están involucrados.

2.2 Modelo de dominio

En la presente investigación no se realizará un modelo de negocio debido a que no se tienen bien definidos los procesos del negocio, es decir, no se tiene una estructura definida de los procesos del negocio (fronteras bien establecidas, donde se logren ver claramente quiénes son las personas que lo inician, quiénes son los beneficiados, pero además quiénes son las personas que desarrollan las actividades en cada uno de estos procesos), por tanto se hace necesario describir el problema a través de un modelo de dominio, también conocido como mapa conceptual.

2.2.1 Análisis de los conceptos del dominio

Se identificaron los principales conceptos presentes en el contexto del problema, los cuales se utilizarán en la confección del modelo de dominio. Dichos conceptos se muestran a continuación:

Colección El Navegante: colección de software educativos que representan hiperentornos de enseñanza-aprendizaje orientada a apoyar la educación secundaria.

Producto multimedia: sistema informático que combina diversos tipos de medias.

Módulo: se le denomina al conjunto de elementos fundamentales que componen el producto.

Contenido: módulo en el que se representan los contenidos de las asignaturas o temas a las cuales corresponde el producto.

Ejercicios: módulo que permite a los estudiantes ejercitar y comprobar los conocimientos y habilidades adquiridas a través de baterías de preguntas o cuestionarios que emiten un criterio a cada una de las respuestas.

Mediateca: módulo responsable de gestionar todo lo referente a los recursos multimedia.

Juegos: módulo que contiene los juegos que los estudiantes tienen a su disposición para ejercitar el contenido aprendido, y a la vez divertirse mientras emplean sus habilidades y conocimientos.

Resultados: módulo que visualiza la traza de la interacción del estudiante con los restantes módulos de la aplicación.

Maestro: módulo que permite al profesor configurar opciones del producto.

Media: se utiliza para referirse a los textos, imágenes, sonidos, videos, animaciones o diaporamas.

Imagen: representación visual que manifiesta la apariencia visual de un objeto real o imaginario.

Video: es la tecnología de la captación, grabación, procesamiento, almacenamiento, transmisión y reconstrucción por medios electrónicos digitales o analógicos de una secuencia de imágenes que representan escenas en movimiento.

Diaporama: sucesión de imágenes o de documentos conectados por efectos y, en los cuales es posible poner sonido.

Animación: es un proceso utilizado para dar la sensación de movimiento a imágenes o dibujos.

Texto: es una composición de signos codificado en un sistema de escritura (como un alfabeto) que forma una unidad de sentido.

Sonido: media que brinda información sonora.

Rompecabezas: juego donde el estudiante organiza las partes de una imagen, usando la lógica hasta lograr organizarla.

Crucigrama: juego donde se muestra en la pantalla un crucigrama. El usuario debe seleccionar cada uno de los números que están en los extremos del crucigrama para responder las preguntas que se le van indicando. La respuesta que el usuario inserte en la caja de texto se podrá ver según el número que eligió para la pregunta en el crucigrama.

Ludo: juego que funciona con preguntas y respuestas, el jugador avanzará según responda las preguntas correspondientes para cada casilla. Se mostrará un tablero de forma cuadrada con tantas fichas como jugadores (hasta 4 jugadores) que se encuentran en las esquinas del tablero en la posición inicial, según el color (amarillo, azul, rojo o verde). El usuario hace clic en los dados y para poder sacar las fichas de su posición inicial será necesario obtener un par de dígitos iguales. Se debe destacar, que en el caso de que se encuentre una ficha de otro jugador en la celda de salida y el propietario de la misma obtenga un par de dígitos iguales, esto le permitirá sustituir su lugar y enviar la ficha del otro jugador a la posición inicial. Es necesario recorrer todas las celdas del tablero evitando ser suplantado por otro jugador y enviado de nuevo a la posición inicial.

Descubre la imagen: juego donde se mostrarán en la pantalla un grupo de casillas en blanco donde aparecerán las letras que el usuario vaya seleccionando hasta adivinar la frase relacionada al tema que eligió para el juego, a medida que vaya acertando en letras se irán mostrando pedazos de la imagen a descubrir. Es parecido al ahorcado, pero con la diferencia que se irán mostrando pistas en la imagen a medida que se van descubriendo letras.

Sopa de letras: juego donde se mostrará en la pantalla una sopa de letras, el usuario selecciona la primera y la última letra de la palabra que encontró dentro de la sopa en cualquier sentido, la misma se marcará en otro color y se puede seleccionar la opción revisar para cerciorarse de que fue correcta la respuesta dada.

2.2.2 Diagrama del modelo de dominio

El modelo del dominio se realiza con el propósito de capturar los objetos o clases más importantes dentro del contexto del sistema y las relaciones entre ellos. Este mapa conceptual representa conceptos del mundo real significativos para un problema, no de los componentes de software. Se describe mediante diagramas de UML, específicamente mediante diagramas de clases. En la figura 2.1 se muestra el Diagrama del modelo de dominio del módulo Juegos de la colección El Navegante.

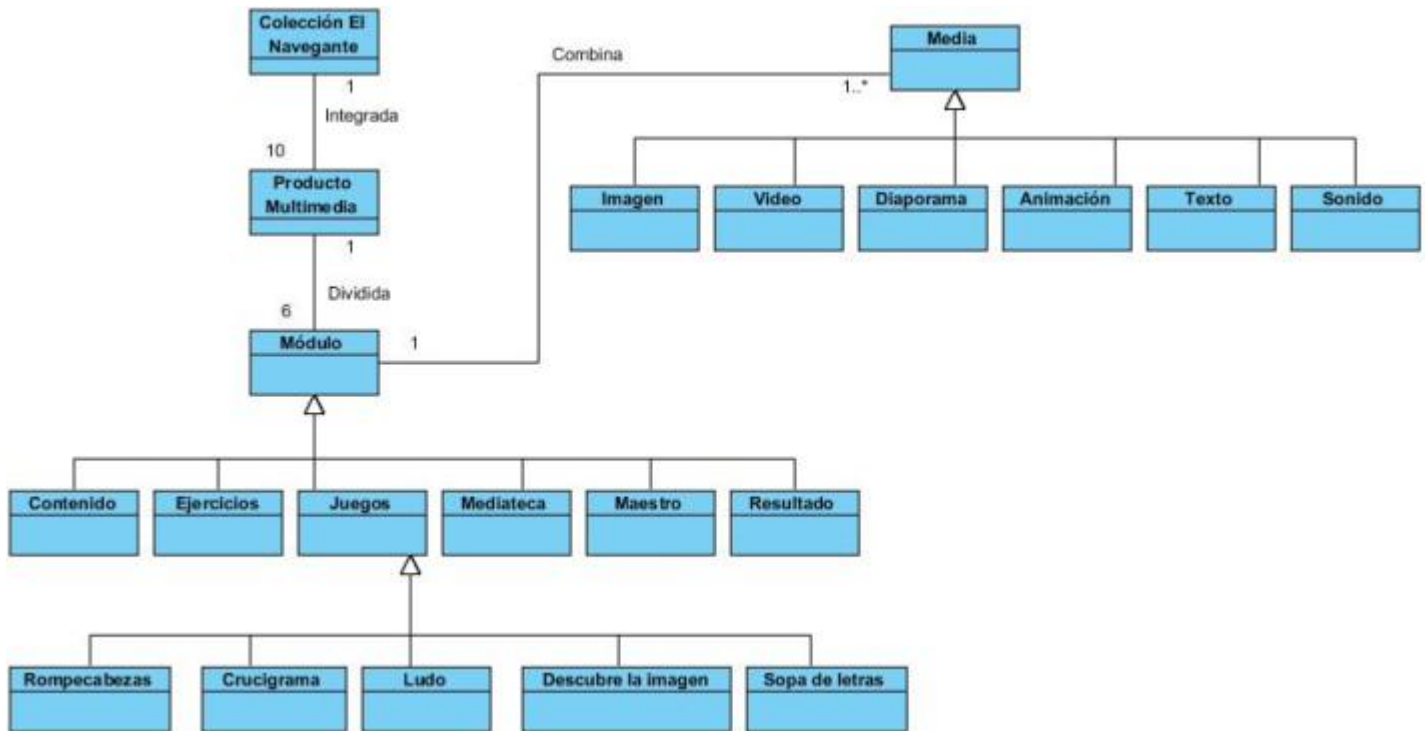


Figura 2.1 Modelo de dominio.

2.3 Descripción del sistema propuesto

El módulo Juegos de la colección El Navegante agrupa un conjunto de juegos que los estudiantes tienen a su disposición para ejercitar el contenido aprendido. El módulo está compuesto por: Descubre la imagen, Crucigrama, Sopa de letras, Ludo y Rompecabezas. Algunos de estos juegos constan de sistemas de preguntas y respuestas, otros se basan en la agilidad mental. Una vez que el usuario selecciona uno o varios juegos, estos son cargados y el sistema permite la interacción con los mismos. El módulo permite la navegación entre los juegos sin necesidad de concluir el que se está jugando. Después de terminar de jugar se muestran los resultados en orden descendente según la efectividad alcanzada por los jugadores. Además el sistema registra los datos de la interacción del estudiante con el módulo, guardando diferentes trazas para cada juego, esto permite verificar posteriormente los resultados obtenidos por cada estudiante. El sistema consta también de un módulo administrativo denominado Backend para gestionar los juegos, ya sea para adicionarlos, eliminarlos, listarlos o modificarlos. El módulo Juegos permite que los estudiantes demuestren los conocimientos adquiridos de una manera diferente, a la vez que el profesor puede llevar un control de lo que fueron capaces de realizar sus alumnos.

2.4 Especificación de requisitos

Para que un proceso de software sea exitoso es imprescindible una descripción detallada, correcta y exhaustiva de los requisitos, ya que es necesario saber qué debe hacer el sistema, y además tener bien claro las expectativas del cliente con respecto al sistema a desarrollar. Los requisitos expresan lo que los usuarios quieren del producto, estos son especificados por escrito como parte de un documento formal. Los requisitos (o requerimientos) son condiciones que deben estar presentes en el software, estos se clasifican en: funcionales y no funcionales. Los requisitos funcionales definen las acciones que el sistema será capaz de realizar, son capacidades o condiciones que debe cumplir para satisfacer las necesidades funcionales del cliente. Los requisitos no funcionales describen aquellas propiedades o cualidades del sistema una vez que esté finalizado, como por ejemplo, el rendimiento, interfaces de usuario, fiabilidad, mantenimiento, seguridad, portabilidad y estándares.

A continuación se muestran los requisitos funcionales y no funcionales del módulo Juegos de la colección El Navegante.

2.4.1 Requisitos funcionales

- RF 1:** Mostrar la pantalla principal del módulo Juegos.
- RF 2:** Seleccionar temas del juego Descubre la imagen.
- RF 3:** Seleccionar temas del juego Sopa de letras.
- RF 4:** Seleccionar temas del juego Crucigrama.
- RF 5:** Seleccionar temas del juego Ludo.
- RF 6:** Seleccionar el nivel de dificultad del juego Sopa de letras.
- RF 7:** Seleccionar la cantidad de juegos a completar.
- RF 8:** Mostrar los avatares de los jugadores implicados en el juego.
- RF 9:** Calcular efectividad.
- RF 10:** Mostrar los resultados del juego.
- RF 11:** Mostrar el tiempo consumido por el jugador en el juego.
- RF 12:** Navegar entre las actividades de cada juego.
- RF 13:** Revisar la respuesta dada por el jugador.
- RF 14:** Mostrar el resultado correcto en el juego Crucigrama.
- RF 15:** Mostrar el resultado correcto en el juego Sopa de letras.

RF 16: Limitar tiempo en el juego Descubre la imagen.

RF 17: Comenzar juego.

RF 18: Registrar la traza de los estudiantes en el módulo Juegos.

RF 19: Registrar la traza de los estudiantes en el juego Sopa de letras.

RF 20: Registrar la traza de los estudiantes en el juego Ludo.

RF 21: Registrar la traza de los estudiantes en el juego Rompecabezas.

RF 22: Registrar la traza de los estudiantes en el juego Crucigrama.

RF 23: Registrar la traza de los estudiantes en el juego Descubre la imagen.

RF 24: Gestionar juego.

RF 24.1: Modificar juego.

RF 24.2: Eliminar juego.

RF 24.3: Ver juego.

RF 24.4: Adicionar juego.

RF 25: Listar juegos.

2.4.2 Requisitos no funcionales

RNF 1: Requisitos de usabilidad.

- Diseño sencillo y estándar por lo que no requiere de previa preparación la utilización del módulo por parte de los estudiantes u otros usuarios sin avanzados conocimientos de computación.

RNF 2: Requisitos de hardware.

Uso local:

- Requerimientos mínimos: 256 MB RAM, 9 GB de HDD, 500 MHz.
- Tiempo de respuesta de una página: no mayor a 30 segundos.

Uso en red:

- Velocidad de conexión mínima: 10 Mbit/s.
- Requerimientos mínimos de hardware: 128 MB RAM, 9 GB de HDD, 500 MHz.
- Velocidad de respuesta máxima: 5 minutos (sobre todo para el caso de las páginas que posean grandes cargas de recursos multimediales).

RNF 3: Requisitos de software.

- Computadora personal con navegador Mozilla Firefox versión 3.5 o cualquier otra versión compatible con éste.
- Sistema operativo Ubuntu en su versión 10.0.4.
- Se podrá utilizar también en los sistemas operativos Linux, Mac y Windows.

RNF 4: Requisitos de restricciones en el diseño y la implementación.

- Framework del lado del cliente: JQuery 1.5.
- Framework del lado del servidor: Symfony 1.4.3.
- Lenguaje de programación del lado del cliente: JavaScript.
- Lenguaje de programación del lado del cliente: CSS.
- Lenguaje de programación del lado del servidor: PHP 5.2.
- Servidor web: Apache
- Patrón arquitectónico: Modelo-Vista-Controlador.
- IDE de desarrollo: NetBeans 6.9.
- Servidor de base de datos: PostgreSQL 8.4.

RNF 5: Requisitos Legales.

- Deben respetarse las licencias y condiciones legales bajo las que fueron adquiridos cada uno de los recursos multimediales.
- Debe mostrarse de cada media la fuente de donde fue extraída.

RNF 6: Requisitos de soporte.

- Se realizará transferencia tecnológica de la colección a los clientes.
- Se impartirán clases a los profesores para explicar el funcionamiento y utilidad del producto.

RNF 7: Requisitos de apariencia e interfaz externa.

- Las interfaces de usuario estarán acorde a las edades a las que van dirigidos cada uno de los 10 productos que integran la colección El Navegante.
- Las interfaces se desarrollarán teniendo en cuenta elementos de las aplicaciones con tecnología multimedia diseñadas para escritorio y las aplicaciones web.
- El diseño de la interfaz gráfica deberá garantizar la distinción visual entre los elementos del sistema.

2.5 Modelo de casos de uso del sistema

El modelo de casos de uso del sistema representa gráficamente la interacción de los casos de uso con los actores del sistema. Este modelo le permite a los desarrolladores y al cliente llegar a un acuerdo sobre los requisitos que debe presentar el producto. Los casos de uso del sistema son un conjunto de secuencias de acciones que un sistema ejecuta y que produce un resultado observable para un actor, es decir, fragmentos de funcionalidad que el sistema ofrece a los actores que interactúan con el mismo. (32)

2.5.1 Actores del sistema

Un actor es una persona, sistema o dispositivo que interactúa con el sistema, iniciando, recibiendo los resultados o participando en alguna de las acciones de un caso de uso. Representa un rol. (32)

Actor	Descripción
Estudiante	Estudiante de escuela secundaria, el cual interactúa con el software, puede jugar con los diversos juegos del módulo, ver los resultados de estos y las trazas de los juegos.
Usuario del Frontend	Persona que interactúa con el software, puede jugar con los diversos juegos del módulo y ver los resultados obtenidos de estos.
Usuario del Backend	Persona que interactúa con el software para gestionar los juegos, adicionarlos y listarlos.

2.5.2 Diagrama de casos de uso del sistema

A continuación se presenta el diagrama de casos de uso del sistema, este muestra la relación entre los actores y los casos de uso, quedando representadas gráficamente las funcionalidades del sistema y su interacción con los usuarios.

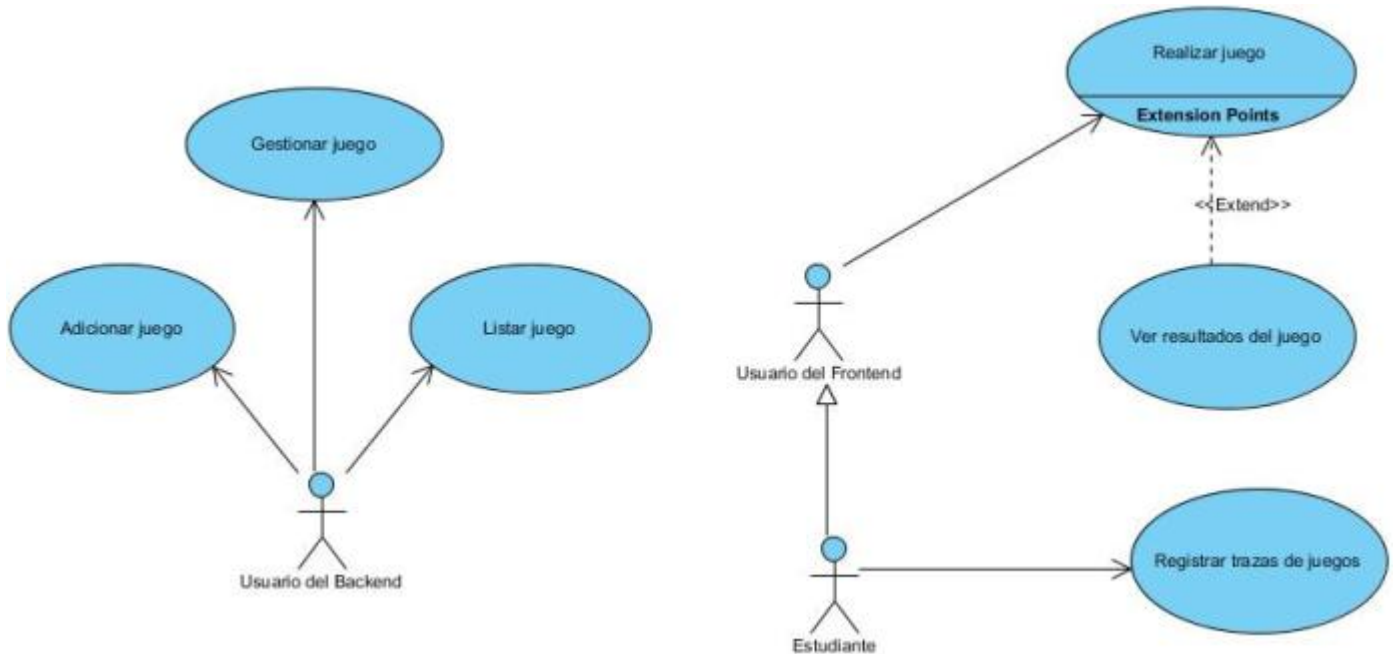


Figura 3 Diagrama de casos de uso del sistema.

2.5.3 Descripción de los casos de uso del sistema.

En el siguiente epígrafe se realiza una descripción de los casos de uso presentes en el diagrama de casos de uso. La descripción de un caso de uso detalla una secuencia de interacciones usuario-sistema, incluyendo cómo comienza y cómo termina. Estas descripciones tienen como objetivo mostrarle al cliente un acercamiento de la propuesta de sistema y orientar a los desarrolladores durante la implementación. A continuación se muestran las descripciones de los CU arquitectónicamente significativos de la propuesta para el módulo Juegos. Para consultar las restantes descripciones ver los anexos, sección Anexo 1 Descripciones textuales de los CU.

Caso de Uso	Realizar juego
Actor	Usuario de Frontend
Resumen	El caso de uso se inicia cuando el actor accede al módulo Juegos, selecciona uno de los siguientes juegos: Sopa de letras, Crucigrama, Descubre la imagen, Ludo o Rompecabezas. El sistema carga el juego seleccionado y permite la interacción con el mismo. El usuario desarrolla las actividades en el juego y finaliza así el caso

	de uso.
Precondiciones	Se ha generado el escritorio de trabajo del usuario autenticado.
Poscondiciones	Se cargó el juego seleccionado por el usuario. El usuario interactuó con el juego.
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1.- El caso de uso se inicia cuando el actor accede al módulo Juegos.	
	<p>2.- Muestra las imágenes que representan a los juegos: Sopa de letras, Crucigrama, Descubre la imagen, Ludo y Rompecabezas.</p> <p>3.- Si selecciona el juego Sopa de letras, ir a la sección 1: "Sopa de letras".</p> <p>Si selecciona el juego Descubre la imagen, ir a la sección 2: "Descubre la imagen".</p> <p>Si selecciona el juego Crucigrama, ir a la sección 3: "Crucigrama".</p> <p>Si selecciona el juego Ludo, ir a la sección 4: "Ludo".</p>
4.- Selecciona el juego: Rompecabezas.	
	<p>5.- Muestra una interfaz para la selección de rompecabezas.</p> <p>Permite:</p> <ul style="list-style-type: none"> • Seleccionar la cantidad rompecabezas. • Seleccionar el tipo de corte para las piezas: Recto, Curvo o Irregulares. • Aceptar. • Cancelar.
6 - Selecciona la cantidad de Rompecabezas, el número de piezas, el tipo de corte de piezas y elige la opción Aceptar.	

	<p>7 - Carga el juego.</p> <p>8 - Muestra una interfaz con los siguientes elementos:</p> <ul style="list-style-type: none"> • Avatar de cada uno de los jugadores implicados en el juego. • El avatar del jugador en turno resaltado. • Nombre del jugador en turno. • Ubicación. • Efectividad 0% • Tiempo transcurrido. • Nivel de ayuda En blanco. <p>Permite:</p> <ul style="list-style-type: none"> • Seleccionar un nivel de ayuda: En blanco, Imagen total e Imagen atenuada. • Realizar las actividades del juego. • Regresar al menú de Juegos. • Seleccionar la opción Siguiente para ir al siguiente juego si no es el último. <p>Seleccionar la opción Anterior para ir al juego anterior si no es el primero.</p>
<p>9 - Selecciona un nivel de ayuda.</p> <p>Interactúa con el juego y realiza las actividades del mismo.</p>	
	<p>10 - Finaliza el juego.</p> <p>11 - Si el juego finalizado no es el último que se va a jugar.</p> <p>12 - Finaliza el caso de uso.</p>
<p>Flujo alterno</p>	
<p>6. a Selecciona la opción Cancelar</p>	
<p>Acciones del actor</p>	<p>Respuesta del sistema</p>

	<p>6. a.1 Muestra la pantalla principal del módulo Juegos.</p> <p>6. a.2 Finaliza el caso de uso.</p>
6. b No hay rompecabezas seleccionados.	
Acciones del actor	Respuesta del sistema
	<p>6. b.1 Muestra el siguiente mensaje: “No hay Rompecabezas seleccionados.”</p> <p>6. b.2 Regresa al punto 5 del flujo básico.</p>
6. c La cantidad de rompecabezas no es suficiente para todos los jugadores.	
Acciones del Actor	Respuesta del sistema
	<p>6. c.1 Muestra el siguiente mensaje: “La cantidad de rompecabezas seleccionados no es suficiente para todos los jugadores. Por favor seleccione nuevamente la cantidad.”</p> <p>6. c.2 Regresa al punto 5 del flujo básico.</p>
9. a Selecciona la opción Siguiente.	
Acciones del Actor	Respuesta del sistema
	<p>9. a.1 Carga el siguiente juego.</p> <p>9. a.2 Regresa al punto 8 del flujo básico.</p>
9. b Selecciona la opción Anterior.	
Acciones del Actor	Respuesta del Sistema
	<p>9. b.1 Carga el juego anterior.</p> <p>9. b.2 Regresa al punto 8 del flujo básico.</p>
9. c Selecciona la opción de regresar a la pantalla principal del módulo Juegos.	
Acciones del Actor	Respuesta del Sistema
	<p>9. c.1 Muestra la pantalla principal del módulo Juegos.</p> <p>9. c.2 Finaliza el caso de uso.</p>
9. d Selecciona el nivel de ayuda Imagen total.	
	<p>9. d.1 Muestra la imagen total que debe obtenerse al unir las piezas del rompecabezas.</p> <p>9. d.2 Regresa al punto 8 del flujo básico.</p>

9. e Selecciona el nivel de ayuda Imagen atenuada.	
Acciones del Actor	Respuesta del Sistema
	9. e.1 Muestra de forma tenue la imagen del rompecabezas a armar. 9. e.2 Ejecuta el punto 8 del flujo básico.
9. f Selecciona el nivel de ayuda En blanco.	
Acciones del actor	Respuesta del sistema
	9. f.1 Muestra en blanco el área para armar el rompecabezas. 9. f.2 Ejecuta el punto 8 del flujo básico.
10. a El juego finalizado es el último.	
Acciones del actor	Respuesta del sistema
	10.a.1 Permite: <ul style="list-style-type: none"> • Aceptar
10. a.2 Selecciona ver los resultados de los juegos.	
	10. a.3 Muestra los resultados del juego. Ver CU Ver Resultados del Juego. 10. a.4 Finaliza el caso de uso.
Sección 1: “Sopa de letras”	
Flujo básico	
Acciones del Actor	Respuesta del Sistema
1 - Selecciona el juego: Sopa de letras	
	2 - Muestra una interfaz con los títulos de los temas que contienen sopas de letras. y permite: <ul style="list-style-type: none"> • Seleccionar uno o varios temas. • Seleccionar la cantidad de sopas de letras a realizar.

	<ul style="list-style-type: none"> • Seleccionar el nivel de dificultad para el juego: Fácil, Medio y Avanzado. • Aceptar • Cancelar
3 - Selecciona uno o varios temas, la cantidad de sopas de letras a realizar, el nivel del juego y elige la opción Aceptar.	.
	<p>4 - Carga el juego.</p> <p>5 - Muestra una interfaz con los siguientes elementos:</p> <ul style="list-style-type: none"> • Enunciado con las instrucciones del juego. • Avatar de cada uno de los jugadores implicados en el juego. • El avatar del jugador en turno resaltado. • Nombre del jugador en turno. • Tiempo • Ubicación • Efectividad en 0% <p>Permite:</p> <ul style="list-style-type: none"> • Realizar las actividades del juego. • Regresar a la pantalla principal del módulo Juegos. • Seleccionar la opción Siguiente para ir al siguiente juego si no es el último. • Seleccionar la opción Anterior para ir al juego anterior si no es el primero.
6 - Interactúa con el juego y realiza las actividades del mismo.	
	7 - Permite revisar la respuesta dada por el jugador.

8 -Selecciona la opción Revisar.	<p>9 - Se muestra dos sopas de letras una con las respuestas correctas y otra con las respuestas dadas por el jugador, señala con tres colores diferentes las palabras correctas, las palabras incorrectas y las palabras que debían encontrarse y no se hallaron. Las palabras correctas se señalan con verdes, las palabras incorrectas se señalan con violeta y las palabras que no se encontraron de color morado.</p> <p>10 - Asigna y muestra al jugador la efectividad obtenida en la sopa de letras.</p> <p>11 - Finaliza el juego.</p> <p>12 - Si el juego finalizado no es el último que se va a jugar finaliza el caso de uso.</p>
Flujo alterno	
3. a Selecciona la opción Aceptar y no ha seleccionado temas.	
Acciones del actor	Respuesta del sistema
	<p>3. a.1 Muestra el siguiente mensaje: “Debe seleccionar al menos un tema para jugar.”</p> <p>3. a.2 Regresa al punto 2 del flujo básico.</p>
3. b Selecciona la opción Cancelar.	
Acciones del actor	Respuesta del sistema
	<p>3. b.1 Muestra la pantalla principal del módulo Juegos.</p> <p>3. b.2 Finaliza el caso de uso.</p>
3. c Selecciona el nivel de dificultad Fácil.	
Acciones del Actor	Respuesta del sistema
	<p>3. c.1 Carga el juego.</p> <p>3. c.2 Muestra las palabras que deben encontrarse en la sopa de letras</p> <p>3. c.3 Ejecuta el punto 5 del flujo básico.</p>
3. d Selecciona el nivel de dificultad Medio.	
Acciones del Actor	Respuesta del sistema

	<p>3. d.1 Carga el juego.</p> <p>3. d.2 Muestra la descripción de cada una de las palabras que deben encontrarse en la sopa de letras.</p> <p>3. d.3 Ejecuta el punto 5 del flujo básico.</p>
3. e Selecciona el nivel de dificultad Avanzado.	
Acciones del Actor	Respuesta del Sistema
	<p>3. e.1 Carga el juego</p> <p>3. e.2 Muestra la cantidad de palabras que deben encontrarse en la sopa de letras y el nombre del tema.</p> <p>3. e.2 Ejecuta el punto 5 del flujo básico.</p>
3. f No hay suficientes sopas de letras para todos los jugadores.	
Acciones del Actor	Respuesta del Sistema
	<p>3. f.1 Muestra el siguiente mensaje: “La cantidad de sopas de letras seleccionadas no es suficiente para todos los jugadores.”</p> <p>3. f.2 Regresa al punto 2 del flujo básico.</p>
6. a Selecciona la opción Siguiente.	
	<p>6. a.1 Carga el siguiente juego.</p> <p>6. a.2 Regresa al punto 5 del flujo básico.</p>
6. b Selecciona la opción Anterior.	
Acciones del Actor	Respuesta del sistema
	<p>6. b.1 Carga el juego anterior.</p> <p>6. b.2 Regresa al punto 5 del flujo básico.</p>
12. a El juego finalizado es el último.	
Acciones del Actor	Respuesta del Sistema
	<p>12. a.1 Permite:</p> <ul style="list-style-type: none"> • Aceptar

12. a.2 Selecciona ver los resultados de los juegos.	
	<p>12. a.3 Muestra los resultados de los juegos. Ver CU Ver Resultados del Juego.</p> <p>12. a.4 Finaliza el caso de uso.</p>
<p>Sección 2: “Descubre la imagen”</p> <p>Flujo básico</p>	
Acciones del Actor	Respuesta del Sistema
1 - Selecciona el juego: Descubre la imagen	
	<p>2 - Muestra una interfaz con los títulos de los temas disponibles.</p> <p>Permite:</p> <ul style="list-style-type: none"> • Seleccionar uno o varios temas. • Seleccionar la cantidad de imagen a descubrir. • Seleccionar la opción de jugar con tiempo limitado. • Aceptar • Cancelar
3 - Selecciona uno o varios temas, la cantidad de imágenes a descubrir y elige la opción Aceptar.	.
	<p>4 - Carga el juego.</p> <p>5 - Muestra una interfaz con los siguientes elementos:</p> <ul style="list-style-type: none"> • Enunciado con las instrucciones del juego. • Avatar de cada uno de los jugadores implicados en el juego.

	<ul style="list-style-type: none"> • El avatar del jugador en turno resaltado. • Nombre del jugador en turno. • Tiempo • Oportunidades • Ubicación • Efectividad en 0% <p>Permite:</p> <ul style="list-style-type: none"> • Realizar las actividades del juego. • Regresar a la pantalla principal del módulo Juegos. • Seleccionar la opción Siguiente para ir al siguiente juego si el juego no es el último. • Seleccionar la opción anterior para ir al juego anterior si el juego no es el primero.
6 - Interactúa con el juego y realiza las actividades del mismo.	
	7 - Asigna y muestra al jugador la efectividad obtenida. 8 - Finaliza el juego.
.	9 - Si el juego finalizado no es el último que se va a jugar. Finaliza el caso de uso.
Flujo alterno	
3. a No hay temas seleccionados.	
Acciones del actor	Respuesta del sistema
	<p>3. a.1 Muestra el siguiente mensaje: “Debe seleccionar al menos un tema para jugar.”</p> <p>3. a.2 Regresa al punto 2 del flujo básico.</p>
3. b Selecciona la opción Cancelar.	
Acciones del actor	Respuesta del sistema

	<p>3. b.1 Muestra la pantalla principal del módulo Juegos.</p> <p>3. b.2 Finaliza el caso de uso.</p>
3. c Selecciona la opción de jugar con tiempo limitado.	
Acciones del Actor	Respuesta del sistema
	<p>3. c.1 Carga el juego.</p> <p>3. c.2 Muestra en el campo tiempo, el tiempo límite que tiene el jugador para descubrir la imagen. El conteo del tiempo se realiza de forma regresiva.</p> <p>3. c.3 Ejecuta el punto 5 del flujo básico.</p>
3. d No hay suficientes imágenes para todos los jugadores.	
Acciones del Actor	Respuesta del sistema
	<p>3. d.1 Muestra el siguiente mensaje: “La cantidad seleccionadas no es suficiente para todos los jugadores.”</p> <p>Permite:</p> <ul style="list-style-type: none"> • Aceptar.
3. d.2 Selecciona la opción Aceptar.	
	3. d.3 Regresa al punto 2 del flujo básico.
6. a Selecciona la opción Siguiente.	
Acciones del Actor	Respuesta del Sistema
	<p>6. a.1 Carga el siguiente juego.</p> <p>6. a.2 Regresa al punto 5 del flujo básico.</p>
6. b Selecciona la opción Anterior.	
	<p>6. a.1 Carga el juego anterior.</p> <p>6. a.2 Regresa al punto 5 del flujo básico.</p>
6. c Se agotó el tiempo para descubrir la imagen.	
Acciones del Actor	Respuesta del sistema

	<p>6. c.1 Muestra el siguiente mensaje: “El tiempo de juego terminó para [nombre del jugador].”</p> <p>6. c.2 Regresa al punto 7 del flujo básico.</p>
9. a El juego finalizado es el último.	
Acciones del Actor	Respuesta del Sistema
	<p>9. a.1 Permite:</p> <ul style="list-style-type: none"> • Aceptar
9. a.2 Selecciona ver los resultados de los juegos.	
	<p>9. a.3 Muestra los resultados de los juegos. Ver CU Ver Resultados del Juego.</p> <p>9. a.4 Finaliza el caso de uso.</p>
Sección 3: “Crucigrama”	
Flujo básico	
Acciones del Actor	Respuesta del Sistema
1 - Selecciona el juego: Crucigrama.	
	<p>2 - Muestra una interfaz con los títulos de los temas disponibles.</p> <p>Permite:</p> <ul style="list-style-type: none"> • Seleccionar uno o varios temas. • Seleccionar la cantidad de crucigramas a realizar. • Aceptar • Cancelar
3 - Selecciona uno o varios temas, la cantidad de crucigramas y elige la opción	.

Aceptar.	
	<p>4 - Carga el juego.</p> <p>5 - Muestra una interfaz con los siguientes elementos:</p> <ul style="list-style-type: none"> • Enunciado con las instrucciones del juego. • Avatar de cada uno de los jugadores implicados en el juego. • El avatar del jugador en turno resaltado. • Nombre del jugador en turno. • Tiempo • Leyenda • Ubicación • Efectividad en 0% <p>Permite:</p> <ul style="list-style-type: none"> • Realizar las actividades del juego. • Regresar a la pantalla principal del módulo Juegos. • Seleccionar la opción Siguiente para ir al siguiente juego si el juego no es el último. • Seleccionar la opción Anterior para ir al juego anterior si el juego no es el primero.
6 - Interactúa con el juego y realiza las actividades del mismo.	
	7 - Permite revisar el crucigrama.
8 - Selecciona la opción Revisar.	9 - Muestra dos crucigramas, uno con la respuesta correcta y otro con la respuesta dada por el jugador, señala en dos colores las respuestas correctas y las respuestas incorrectas en el crucigrama. Señala con color verde las respuestas correctas y con color violeta las respuestas incorrectas.
	10 - Asigna y muestra al jugador la efectividad obtenida.

	<p>11 - Finaliza el juego.</p> <p>12 - Si el juego finalizado no es el último que se va a jugar. Finaliza el caso de uso.</p>
Flujo alternativo	
3. a No hay temas seleccionados.	
Acciones del actor	Respuesta del sistema
	<p>3. a.1 Muestra el siguiente mensaje: “Debe seleccionar al menos un tema para jugar.”</p> <p>3. a.2 Regresa al punto 2 del flujo básico.</p>
3. b Selecciona la opción Cancelar.	
Acciones del actor	Respuesta del sistema
	<p>3. b.1 Muestra la pantalla principal del módulo Juegos.</p> <p>3. b.2 Finaliza el caso de uso.</p>
3. c No hay suficientes crucigramas para todos los jugadores.	
Acciones del Actor	Respuesta del sistema
	<p>3. c.1 Muestra el siguiente mensaje: “La cantidad seleccionada no es suficiente para todos los jugadores.”</p> <p>Permite:</p> <ul style="list-style-type: none"> • Aceptar
3. c.2 Selecciona la opción Aceptar.	
	3. c.3 Regresa al punto 2 del flujo básico.
6. a Selecciona la opción Siguiente.	
Acciones del Actor	Respuesta del sistema
	6. a.1 Carga el siguiente juego.

	6. a.2 Regresa al punto 5 del flujo básico.
6. b Selecciona la opción Anterior.	
Acciones del Actor	Respuesta del Sistema
	6. b.1 Carga el juego anterior. 6. b.2 Regresa al punto 5 del flujo básico.
12. a El juego finalizado es el último.	
	12. a.1 Permite: <ul style="list-style-type: none"> • Aceptar
12. a.2 Selecciona ver los resultados de los juegos.	
	12. a.3 Muestra los resultados de los juegos. Ver CU Ver Resultados del Juego. 12. a.4 Finaliza el caso de uso.
Sección 4: “Ludo”	
Flujo básico	
Acciones del Actor	Respuesta del Sistema
1 - Selecciona el juego: Ludo.	
	2 - Muestra una interfaz con los títulos de los temas disponibles. Permite: <ul style="list-style-type: none"> • Seleccionar uno o varios temas. • Aceptar • Cancelar
3 - Selecciona uno o varios temas y elige la opción Aceptar.	.
	4 - Carga el juego. 5 Muestra una interfaz con los siguientes elementos:

	<ul style="list-style-type: none"> • Enunciado con las indicaciones del juego. • Avatar de cada uno de los jugadores implicados en el juego. • El avatar del jugador en turno resaltado. • Nombre del jugador en turno. <p>Permite:</p> <ul style="list-style-type: none"> • Realizar las actividades del juego.
6 - Interactúa con el juego y realiza las actividades del mismo.	
	<p>7 - Finaliza el juego.</p> <p>8 - Permite:</p> <ul style="list-style-type: none"> • Aceptar.
9 - Selecciona la opción Aceptar.	
	<p>10 - Muestra los resultados del juego. Ver <i>CU Ver Resultados del Juego</i>.</p> <p>11 - Finaliza el caso de uso.</p>
Flujo alterno	
3. a No hay temas seleccionados.	
Acciones del actor	Respuesta del sistema
	<p>3. a.1 Muestra el siguiente mensaje: "Debe seleccionar al menos un tema para jugar."</p> <p>3. a.2 Regresa al punto 2 del flujo básico.</p>
3. b Selecciona la opción Cancelar.	
Acciones del actor	Respuesta del sistema
	3. b.1 Muestra la pantalla principal del módulo Juegos.

3. b.2 Finaliza el caso de uso.

Tabla 2.1 Descripción textual del caso de uso Realizar juego.

Caso de Uso	Gestionar juego
Actor	Usuario de Backend
Resumen	El caso de uso se inicia cuando el usuario de Backend accede al Módulo Administrativo Juegos, selecciona la opción Listar juego. El sistema muestra un listado con los juegos y permite modificar, eliminar o ver los contenidos. El usuario de Backend realiza la opción que desea y finaliza así el caso de uso.
Precondiciones	Se ha generado el escritorio de trabajo del usuario autenticado. El usuario debe haberse autenticado como un usuario de Backend. Deben haberse listado los juegos.
Poscondiciones	Se modificó, eliminó o vio el contenido de un juego.
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del Sistema
1.- El caso de uso se inicia cuando el actor accede al Módulo Administrativo Juegos y selecciona la opción Listar juego.	
	<p>2.- Muestra un listado con los contenidos generales de los juegos y permite:</p> <ul style="list-style-type: none"> • Modificar juego • Eliminar juego. • Ver juego • Buscar juego • Regresar a la vista principal.

	<p>3 - Si selecciona la opción Eliminar juego, ir a la sección 1: “Eliminar juego”.</p> <p>Si selecciona la opción Ver juego, ir a la sección 2: “Ver juego”.</p>
4- Selecciona la opción Modificar.	
	<p>5 - Muestra los contenidos del juego permitiendo modificar los datos del juego y además:</p> <ul style="list-style-type: none"> • Guardar • Cancelar
6 - Modifica los datos y selecciona la opción Guardar.	<p>7 - Valida los datos introducidos. Guarda los datos de los contenidos del juego.</p> <p>Muestra un mensaje de información. “El juego ha sido actualizado correctamente.”</p> <p>8 - Finaliza el caso de uso.</p>
Flujo alterno	
5. a Selecciona la opción Cancelar.	
Acciones del actor	Respuesta del sistema
	<p>5. a.1 Muestra la vista anterior.</p> <p>5. a.2 Finaliza el caso de uso.</p>
6. a Existen campos vacíos.	
Acciones del actor	Respuesta del sistema
	<p>6. a.1 Muestra el siguiente mensaje: “Existen campos vacíos.”</p> <p>6. a.2 Regresa al punto 5 del flujo básico.</p>
6. b Existen datos incorrectos.	
Acciones del Actor	Respuesta del sistema

	<p>6. b.1 Muestra el siguiente mensaje: “Existen datos incorrectos.”</p> <p>6. b.2 Regresa al punto 5 del flujo básico.</p>
3. a Selecciona la opción Regresar.	
Acciones del Actor	Respuesta del sistema
	<p>3. a.1 Muestra la pantalla principal.</p> <p>3. a.2 Finaliza el caso de uso.</p>
3. b Selecciona la opción Buscar.	
Acciones del Actor	Respuesta del Sistema
	<p>3. b.1 Muestra la pantalla para buscar juego. Ver “CU Listar juego, sección 1”</p> <p>3. b.2 Regresa al paso 3 del flujo básico.</p>
Sección 1: “Eliminar juego”	
Flujo básico	
Acciones del Actor	Respuesta del Sistema
1 - Selecciona la opción Eliminar de un juego.	
	<p>2- Muestra un mensaje de información: “Se eliminará el juego seleccionado. ¿Desea continuar?”</p> <p>Y permite:</p> <ul style="list-style-type: none"> • Aceptar • Cancelar
3 - Selecciona la opción Aceptar.	
	<p>4 - Muestra un mensaje de información: “El juego seleccionado ha sido eliminado”.</p> <p>5 - Finaliza el caso de uso.</p>
Flujo alterno	
6. b Selecciona la opción Cancelar.	

Acciones del actor	Respuesta del sistema
	6. b.1 Muestra la vista anterior. 6. b.2 Finaliza el caso de uso.
Sección 2: “Ver juego”	
Flujo básico	
Acciones del Actor	Respuesta del Sistema
1 - Selecciona la opción Ver de un juego.	
	2- Muestra los datos de los contenidos del juego seleccionado. Y permite: <ul style="list-style-type: none"> • Aceptar.
2 – Selecciona la opción Aceptar.	
	4 - Finaliza el caso de uso.

Tabla 2.4 Descripción textual del caso de uso Gestionar juegos.

Caso de Uso	Adicionar juego
Actor	Usuario de Backend
Resumen	El caso de uso se inicia cuando el actor accede al módulo administrativo Juegos, selecciona la opción Adicionar juego. El sistema muestra los contenidos de los juegos y permite adicionar los datos del juego. El actor introduce los datos y finaliza el caso de uso.
Precondiciones	Se ha generado el escritorio de trabajo del usuario autenticado. El usuario debe haberse autenticado como un usuario de Backend.
Poscondiciones	Se adicionó el contenido de un juego.
Flujo Normal de Eventos	
Acciones del Actor	Respuesta del Sistema

<p>1 - El caso de uso se inicia cuando el actor accede al Módulo Administrativo Juegos y selecciona la opción Adicionar juego.</p>	
	<p>2.- Muestra los contenidos generales de los juegos permitiendo introducir:</p> <ul style="list-style-type: none"> • Enunciado <p>Y seleccionar:</p> <ul style="list-style-type: none"> • Tipo de juego
<p>3- Introduce el enunciado y selecciona el tipo de juego.</p>	
	<p>4 - Permite introducir los contenidos específicos referentes al tipo de juego seleccionado y permite guardar.</p> <p>En caso que el tipo de juego sea Crucigrama permite introducir:</p> <ul style="list-style-type: none"> • No. Palabra • Palabra • Forma • Columna • Fila • Pregunta <p>En caso que el tipo de juego sea Sopa de letras permite introducir:</p> <ul style="list-style-type: none"> • Palabra • Descripción

	<ul style="list-style-type: none"> Nivel de dificultad <p>En caso que el tipo de juego sea Descubre la imagen permite introducir:</p> <ul style="list-style-type: none"> Palabra a descubrir Imagen Tiempo
5 - Introduce los datos específicos y selecciona la opción Guardar.	
	<p>6 - Valida los contenidos introducidos. Guarda los contenidos. Muestra un mensaje de información: “Los datos fueron introducidos correctamente”.</p> <p>7 - Finaliza el caso de uso.</p>
Flujo alterno	
6. a Existen campos vacíos.	
Acciones del actor	Respuesta del sistema
	<p>6. a.1 Muestra el siguiente mensaje: “Existen campos vacíos.”</p> <p>6. a.2 Regresa al punto 2 del flujo básico.</p>
6. b Existen datos incorrectos.	
Acciones del actor	Respuesta del sistema
	<p>6. b.1 Muestra el siguiente mensaje: “Existen datos incorrectos.”</p> <p>6. b.2 Regresa al punto 2 del flujo básico.</p>

Tabla 2.5 Descripción textual del caso de uso Adicionar juegos.

2.6 Conclusiones

En el capítulo desarrollado se realizó una descripción de la propuesta del sistema para el módulo Juegos de la colección El Navegante, quedando definidos los principales conceptos asociados al dominio del problema. Se realizó el diagrama de clases del dominio para lograr un entendimiento preciso de lo requerido. Se definieron los actores que intervienen en el sistema y se identificaron los casos de uso necesarios para definir las funcionalidades básicas de la aplicación, partiendo de los requisitos funcionales especificados, estas funcionalidades fueron representadas mediante el diagrama de casos de uso del sistema. Finalmente se describieron textualmente los casos de usos definidos en el sistema. De esta forma es posible comenzar una fase posterior donde se realizará el análisis y diseño del sistema propuesto.

Capítulo 3: Análisis y diseño

3.1 Introducción

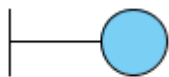
El análisis de un sistema proporciona una visión abstracta del mismo y constituye el punto de partida para el diseño, donde se modela el sistema para que soporte todos los requisitos. El diseño es un refinamiento del análisis que tiene en cuenta también los requisitos no funcionales para representar en un lenguaje técnico la estructura de clases de la aplicación y las relaciones entre ellas. Se hace un análisis del código y se manifiesta el modelo de implementación propuesto, este último se describe mediante los diagramas de componentes y despliegue.

3.2 Modelo de análisis

El modelo de análisis sirve como una primera aproximación del diseño, este ayuda a entender los requisitos funcionales ya que representa una vista interna del sistema descrita en un lenguaje más técnico. Este modelo estructura el funcionamiento interno del sistema mediante las clases del análisis y las asociaciones entre estas. En el modelo del análisis no se toma en cuenta el lenguaje de programación a usar en la construcción, el objetivo del análisis es comprender perfectamente los requisitos del software, es decir se centra en describir qué debe hacer el sistema y no en cómo lo va a hacer. (33)

3.2.1 Diagramas de clases del análisis

El diagrama de clases del análisis tiene como objetivo visualizar la información a través de una representación visual de los objetos, exponiendo al usuario un diseño donde se evidencian las interrelaciones que se producen entre actores y clases. En la construcción del diagrama de clases del análisis se identifican las clases que describirán la realización de los casos de uso y las relaciones entre ellas. Se utilizan tres tipos de clases y se identifican con los siguientes estereotipos:



Cl_Ejemplo

Interfaz: se encargan de manejar la interacción entre el sistema y sus actores.



CC_Ejemplo

Control: coordinan la realización de uno o varios casos de uso controlando las actividades de los objetos que implementan la funcionalidad del caso de uso.



CE_Ejemplo

Entidad: representa la información que generalmente es de larga duración y a menudo persistente.

A continuación se muestran los diagramas de clases del análisis para algunos de los casos de uso del sistema, para consultar los restantes diagramas ver los anexos, sección Anexo 2 Diagramas de clases del análisis.

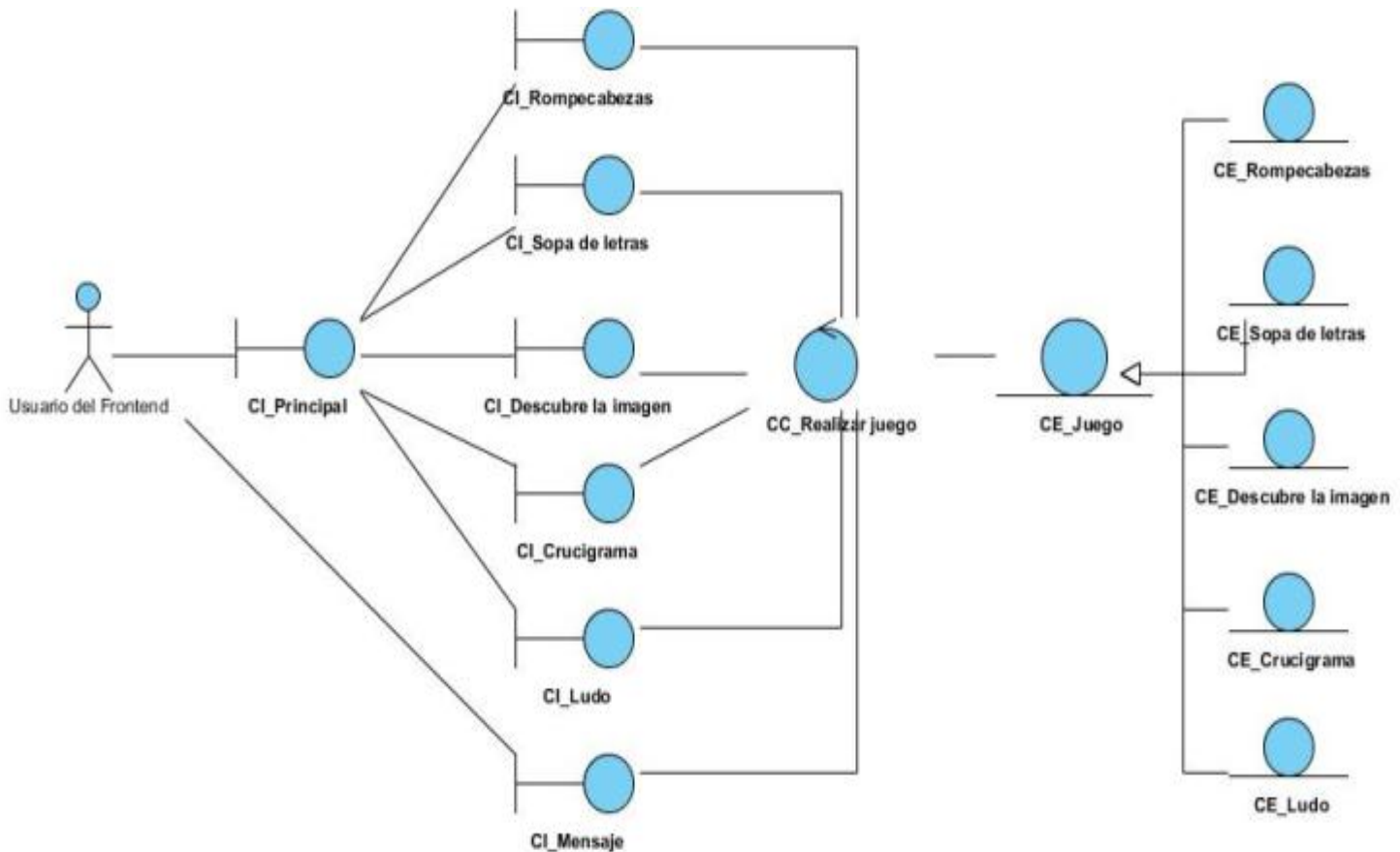


Figura 3.1 Diagrama de clases del análisis del CU Realizar juego.

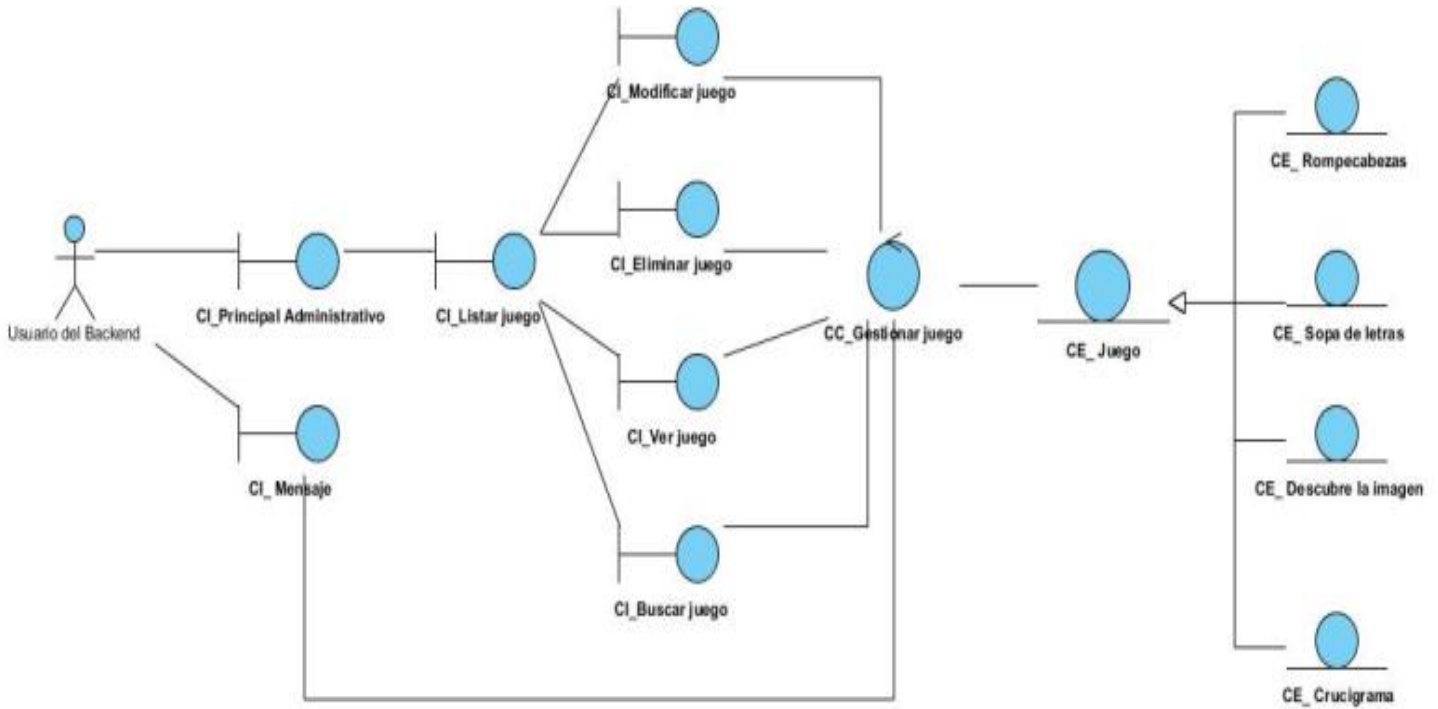


Figura 3.4 Diagrama de clases del análisis del CU Gestionar juego.

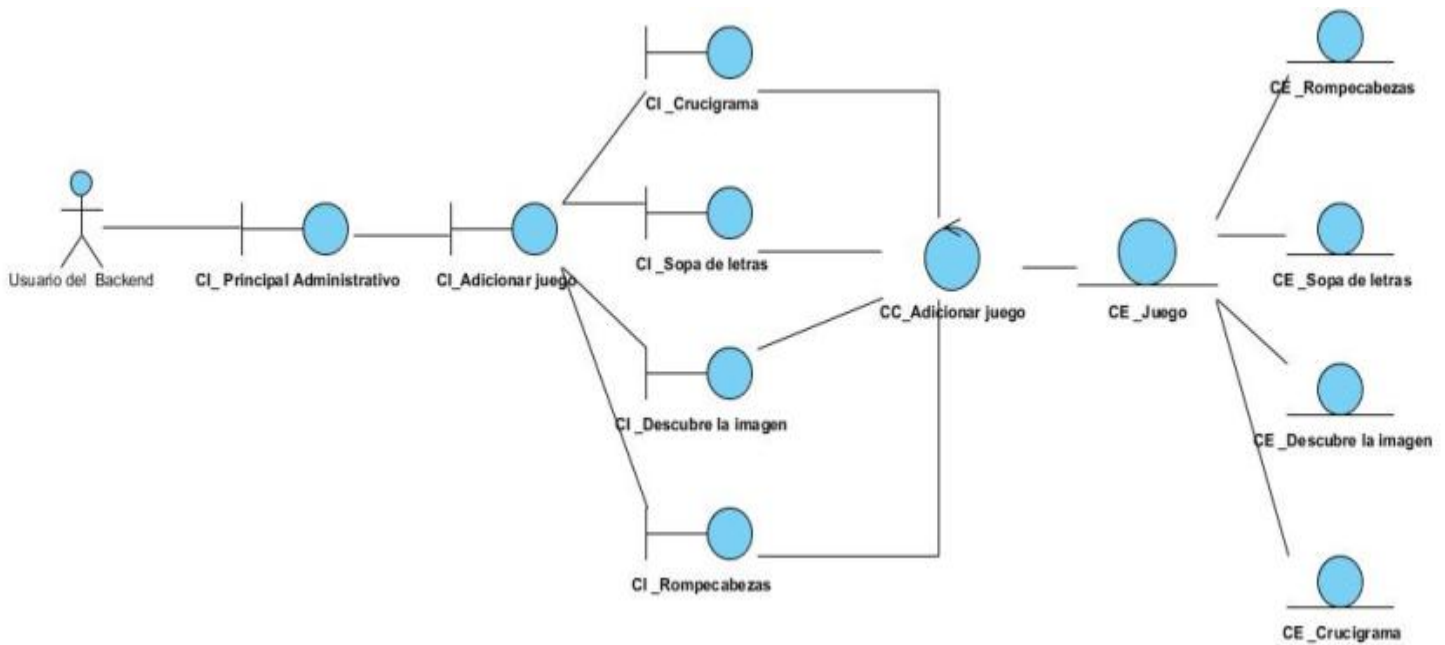


Figura 3.5 Diagrama de clases del análisis del CU Adicionar juego.

3.3 Modelo de diseño

El modelo de diseño es un modelo de objeto que describe las realizaciones de los casos de uso y sirve como una abstracción del modelo de implementación y su código fuente. En el diseño se modela el sistema, incluida la arquitectura, para que soporte todos los requisitos, incluyendo los no funcionales. (34) El modelo de diseño tiene como propósito transformar los requisitos tanto funcionales como no funcionales en un diseño de clases que describa las relaciones entre estas, en un lenguaje técnico lo más cercano posible al lenguaje de programación.

3.3.1 Patrones de diseño

Los patrones permiten aprovechar lo hecho y evitar que se cometan errores que comúnmente retrasan los proyectos. Se puede decir que los patrones de diseño son soluciones exitosas a problemas comunes, que se pueden aplicar a diferentes problemas de diseño en distintas circunstancias.

Se hizo uso del patrón de diseño Modelo Vista Controlador, con este se logra la presencia de otros patrones como:

Bajo acoplamiento: Existe un mínimo de dependencia entre el modelo, la vista y el controlador (un cambio realizado en un componente no implica cambios en otro).

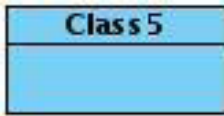
Alta cohesión: Cada elemento del patrón está altamente especializado en su tarea (la vista en mostrar datos al usuario, el controlador en las entradas y el modelo en su objetivo de negocio).

También se utilizó el patrón Mediator, el mediador encapsula el comportamiento de todo un conjunto de objetos en uno solo, este patrón estimula la pérdida de acoplamiento, ya que oculta las referencias explícitas entre los objetos, un objeto no tiene que conocer en detalles cómo interactuar con otros, sino que lo hace a través del mediador.

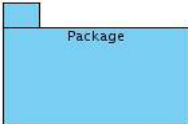
3.3.2 Diagramas de clases del diseño

Los diagramas de clases muestran un conjunto de clases, interfaces y colaboraciones, así como sus relaciones. En el diagrama de clases para las aplicaciones web se modelan las páginas, los enlaces entre estas, el código que irá creando las páginas y el contenido dinámico de estas.

A continuación se muestran las terminologías utilizadas en los diagramas de clases del diseño:



Clases: Una clase es un contenedor de uno o más datos (variables o propiedad miembro) junto a las operaciones de manipulación de dichos datos (métodos).



Paquete: Representa un grupo de elementos (casos de uso, clases, componentes, otros paquetes) relacionados por algún criterio en común.



Server Page [SP]: Representa la página Web que tiene código que se ejecuta en el servidor. Su principal función es construir la página cliente.



Client Page [CP]: Representa una página Web, con formato HTML. Son interpretadas por el navegador, su función es visualizar, interactuar y mostrar lo que el usuario necesita.



Form: Colección de elementos de entrada que son parte de una página cliente. Sus atributos son los elementos de entrada del formulario. Su principal función es enviar los datos entrados por el usuario a la página servidora.

A continuación se muestran los diagramas de clases del diseño para algunos de los casos de uso del sistema, para consultar los restantes diagramas ver los anexos, sección Anexo 3 Diagramas de clases del diseño.

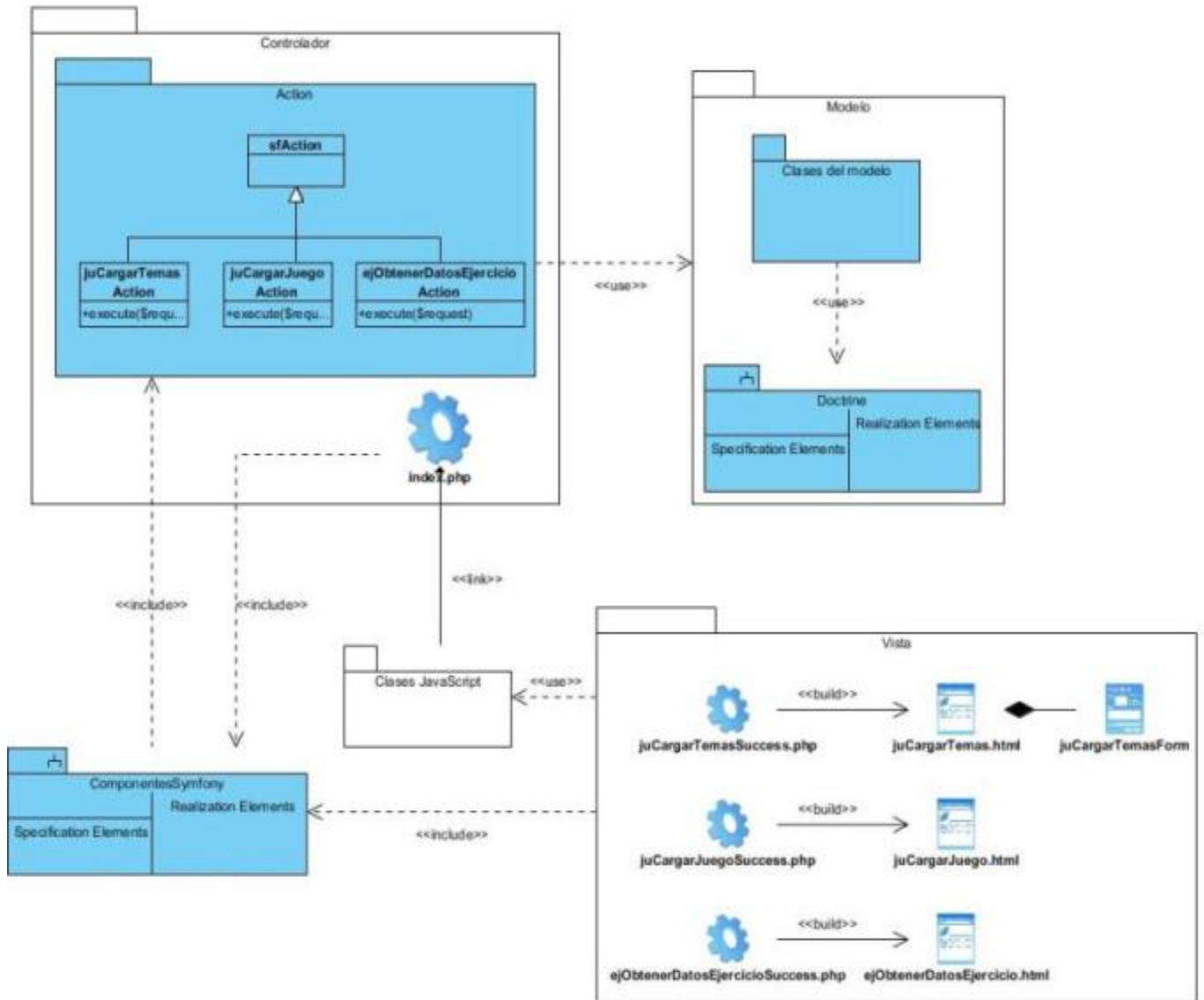


Figura 3.7 Diagrama de clases del diseño del CU Realizar juego.

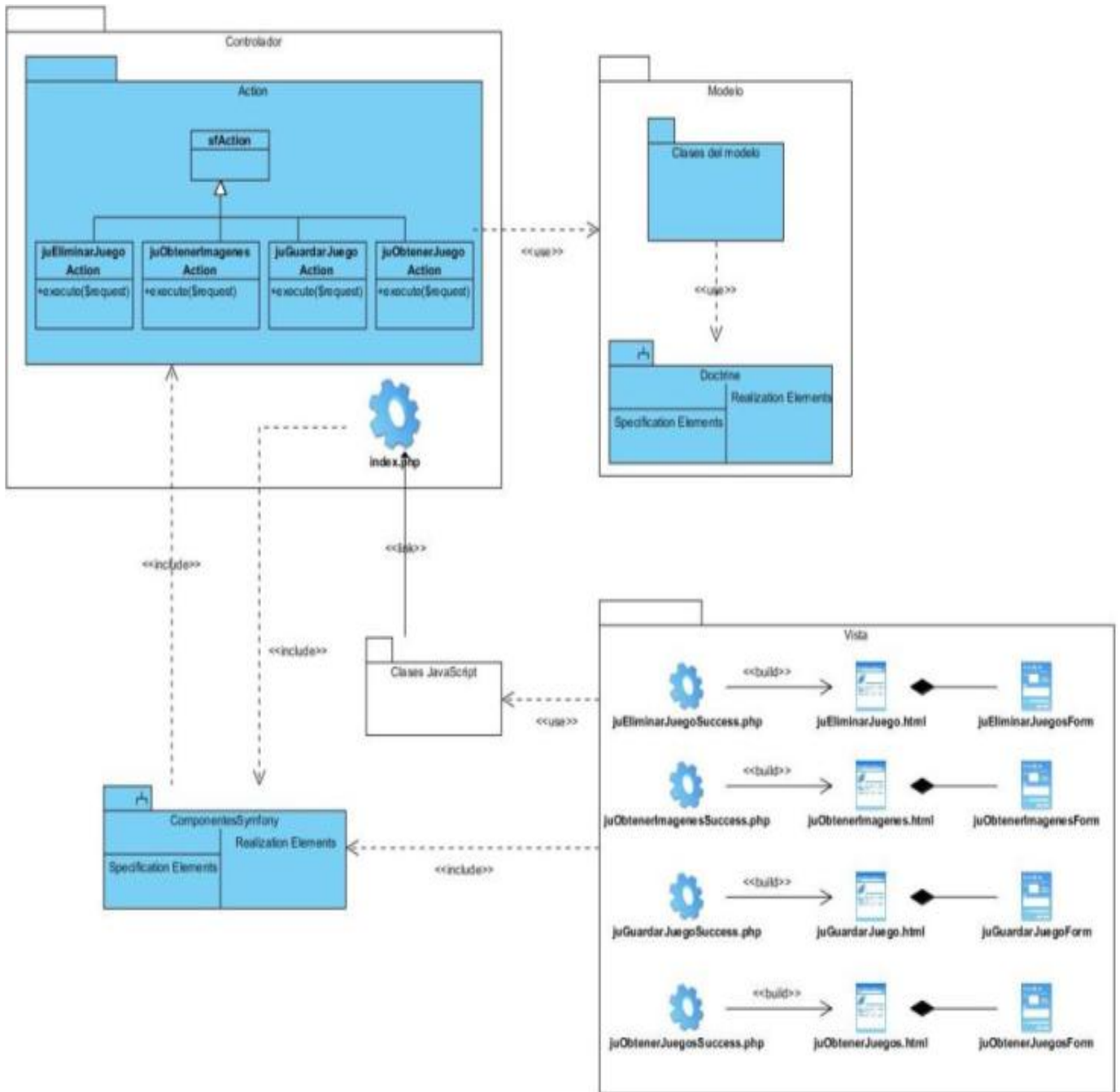


Figura 3.10 Diagrama de clases del diseño del CU Gestionar juego.

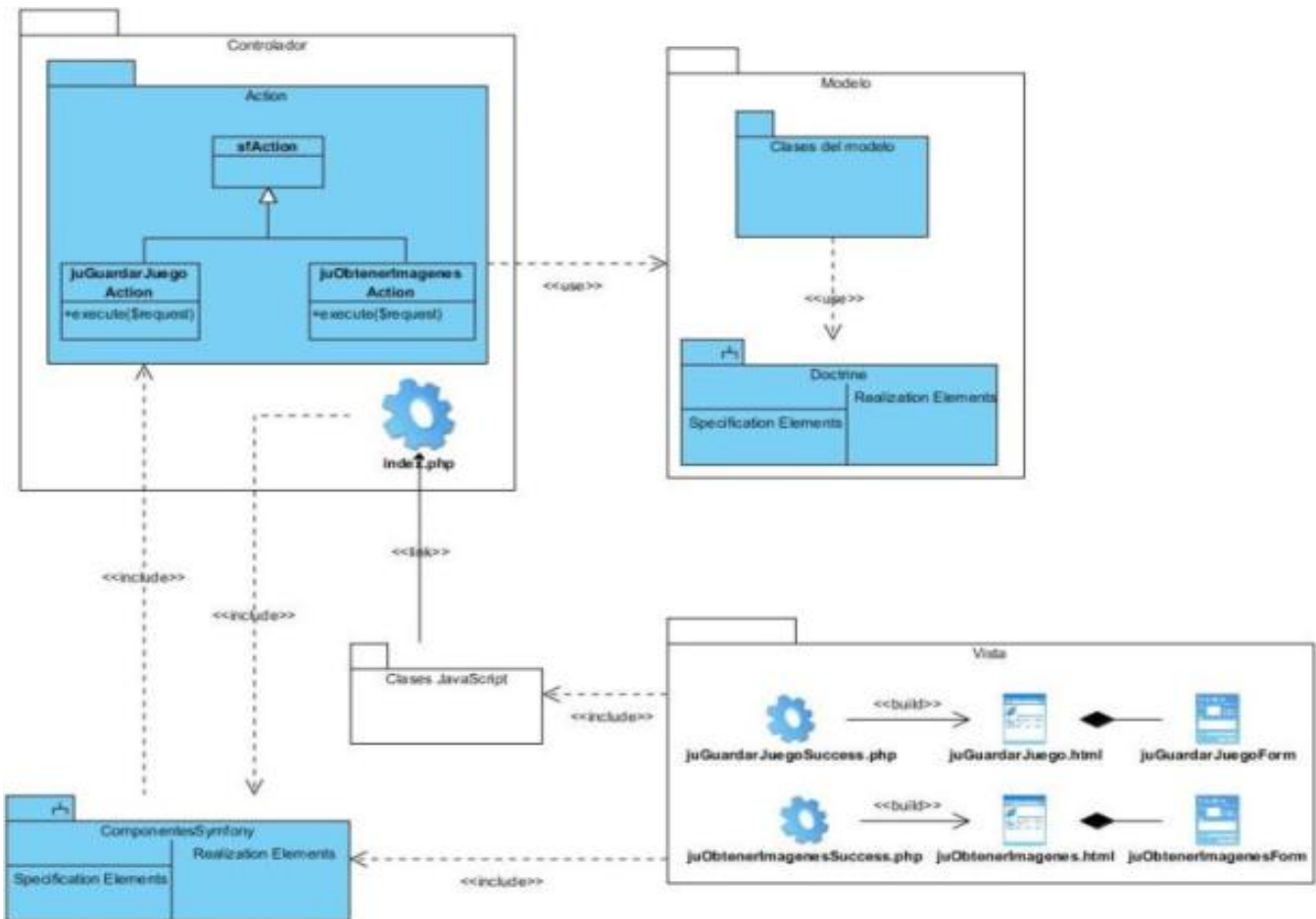


Figura 3.11 Diagrama de clases del diseño del CU Agregar juego.

Los subsistemas de clases javaScript se pueden consultar en los anexos, sección Anexo 4 Subsistemas de clases javaScript.

3.4 Conclusiones

En este capítulo se generaron diferentes artefactos, como resultado del flujo de trabajo análisis y diseño. Se documentaron los diagramas de clases del análisis, proporcionando un mayor entendimiento de los requisitos y se introdujeron los diagramas de clases del diseño, ambos confeccionados para cada caso de uso del sistema.

Capítulo 4: Implementación y prueba

4.1 Estándar de codificación.

Los estándares de codificación son reglas que se siguen para la escritura del código fuente, con estos todos los programadores del proyecto trabajan de forma más organizada. Estos estándares permiten una mejor integración entre las líneas de código y establecen pautas que conlleven a lograr un código más legible y reutilizable.

Los estándares para el desarrollo de la colección han sido definidos por la arquitectura del proyecto Colecciones de Software Multisaber y El Navegante, estos se pueden consultar en los anexos, sección Anexo 5 Estándares de codificación.

4.2 Modelo de implementación

El Modelo de implementación tiene gran importancia ya que mediante su uso los desarrolladores pueden entender claramente el funcionamiento del sistema antes de comenzar a escribir las líneas de código. Este modelo está conformado por los diagramas de componentes y de despliegue, y tiene como objetivo fundamental desarrollar la arquitectura y el sistema como un todo.

4.2.1 Diagrama de componentes

El diagrama de componentes muestra un conjunto de elementos de los modelos como componentes, subsistemas de implementación y sus relaciones. Este diagrama facilita el entendimiento del modelo de implementación, ya que en él se presentan los componentes lógicos de la aplicación así como las relaciones de dependencias que existen entre ellos.

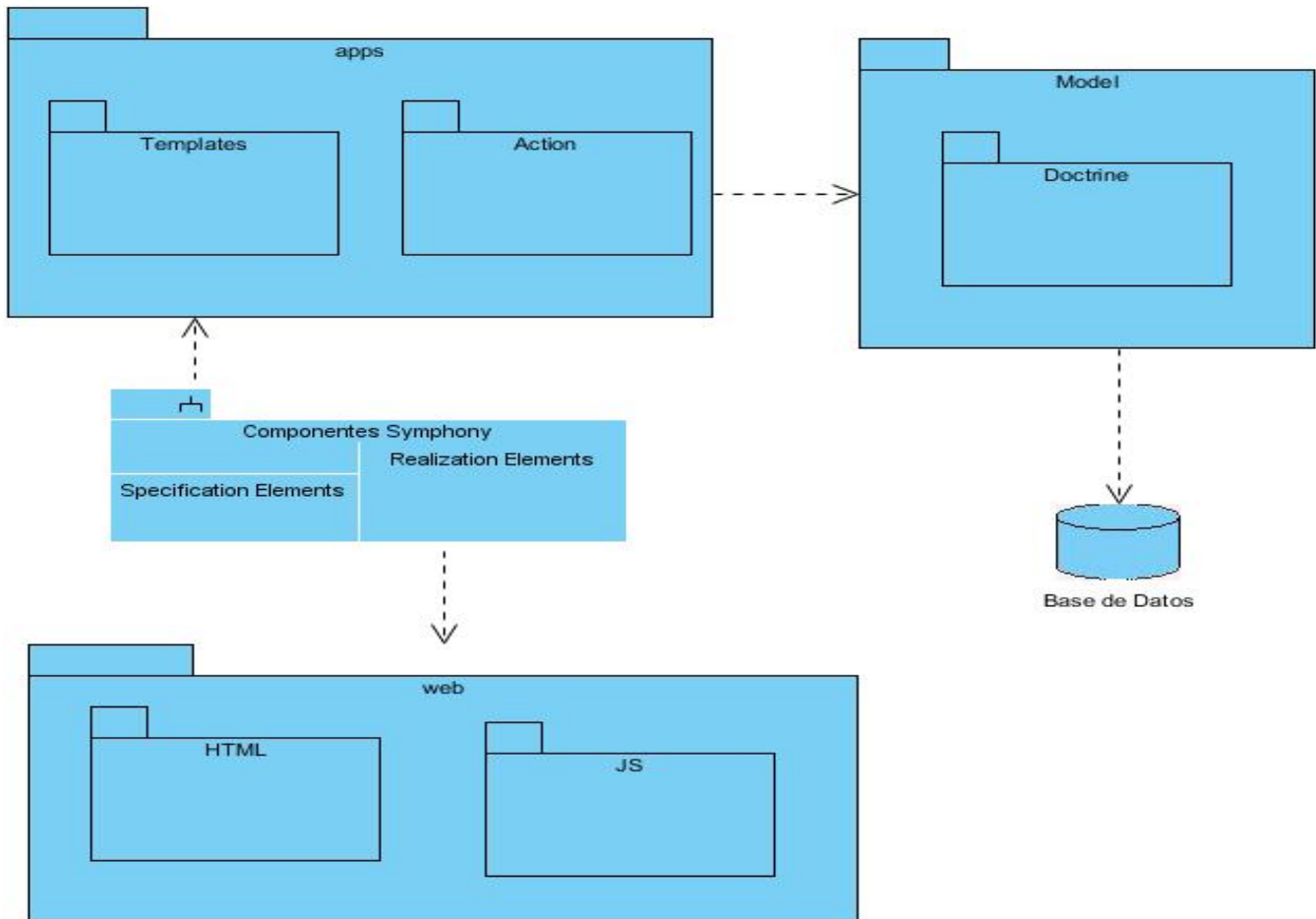


Figura 3.19 Diagrama de componentes.

4.2.2 Diagrama de despliegue

Los diagramas de despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. Un nodo es un recurso de ejecución que puede ser un procesador o un dispositivo de hardware similar (35). A continuación se muestran cómo están distribuidos los nodos y dispositivos para este producto, en este caso están distribuidos en dos variantes.

Variante 1:

Instalar localmente el producto en la PC del usuario el cual va a tener instalado el servidor web y la base de datos. La impresora como dispositivo utilizado por el sistema para posibilitar la impresión de documentos.



Figura 3.20 Diagrama de despliegue variante 1.

Variante 2:

Se necesita de un ordenador que funcione como servidor central, el cual tendrá instalado el servidor web y el servidor de base de datos. Además se necesita de una computadora que haga función de cliente, en el cual el usuario va a poder interactuar con la aplicación y la impresora como dispositivo utilizado por el sistema para posibilitar la impresión de documentos.

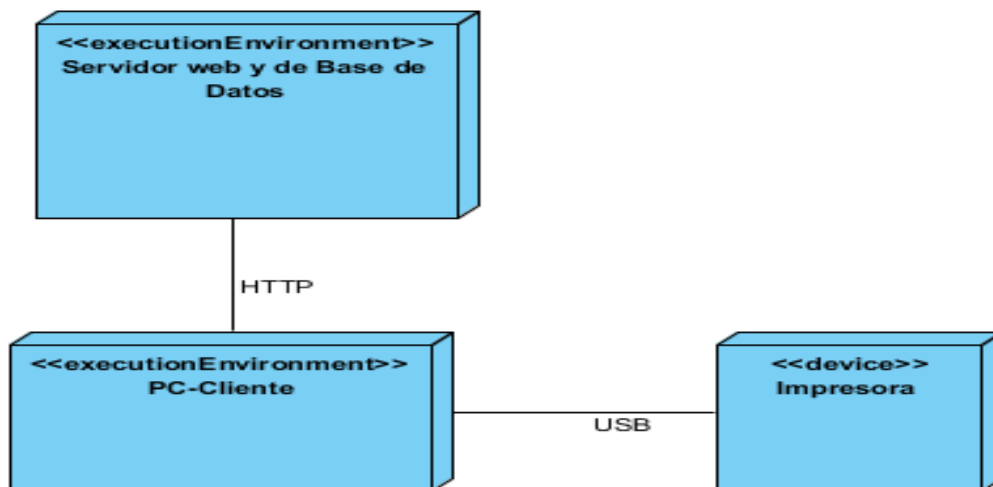


Figura 3.21 Diagrama de despliegue variante 2.

Descripción de la funcionalidad y capacidad de los nodos.

PC Cliente: ordenador cliente que se conecta a través de un navegador web al servidor central donde reside la aplicación. No se necesita tener instalado el producto localmente.

PC Servidor: servidor central, el cual tiene instalado un servidor web, uno 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 va a establecer entre una PC Cliente con el servidor central donde reside el servidor web y el de base de datos, es decir, significa la conexión entre el navegador web y el servidor de aplicaciones.

<<USB>>

Conexión que existe entre la impresora y la PC Cliente del usuario. Se modela con USB, pero puede existir otro tipo, en dependencia del tipo de impresora que esté usando el usuario y el tipo de conexión que utilice ésta.

4.3 Modelos de datos

El modelo de datos se utiliza para describir la estructura lógica y física de la información persistente gestionada por el sistema. Este artefacto describe las representaciones lógicas y físicas de datos persistentes utilizados por la aplicación. (34)

El modelo de datos es una colección de herramientas conceptuales para describir los datos y las relaciones que existen entre ellos, permite representar las estructuras presentes en la base de datos, su tipo y la forma en que se relacionan. De todas las tablas de la base de datos solamente se muestran las tablas que se relacionan con el módulo Juegos de la colección El Navegante.

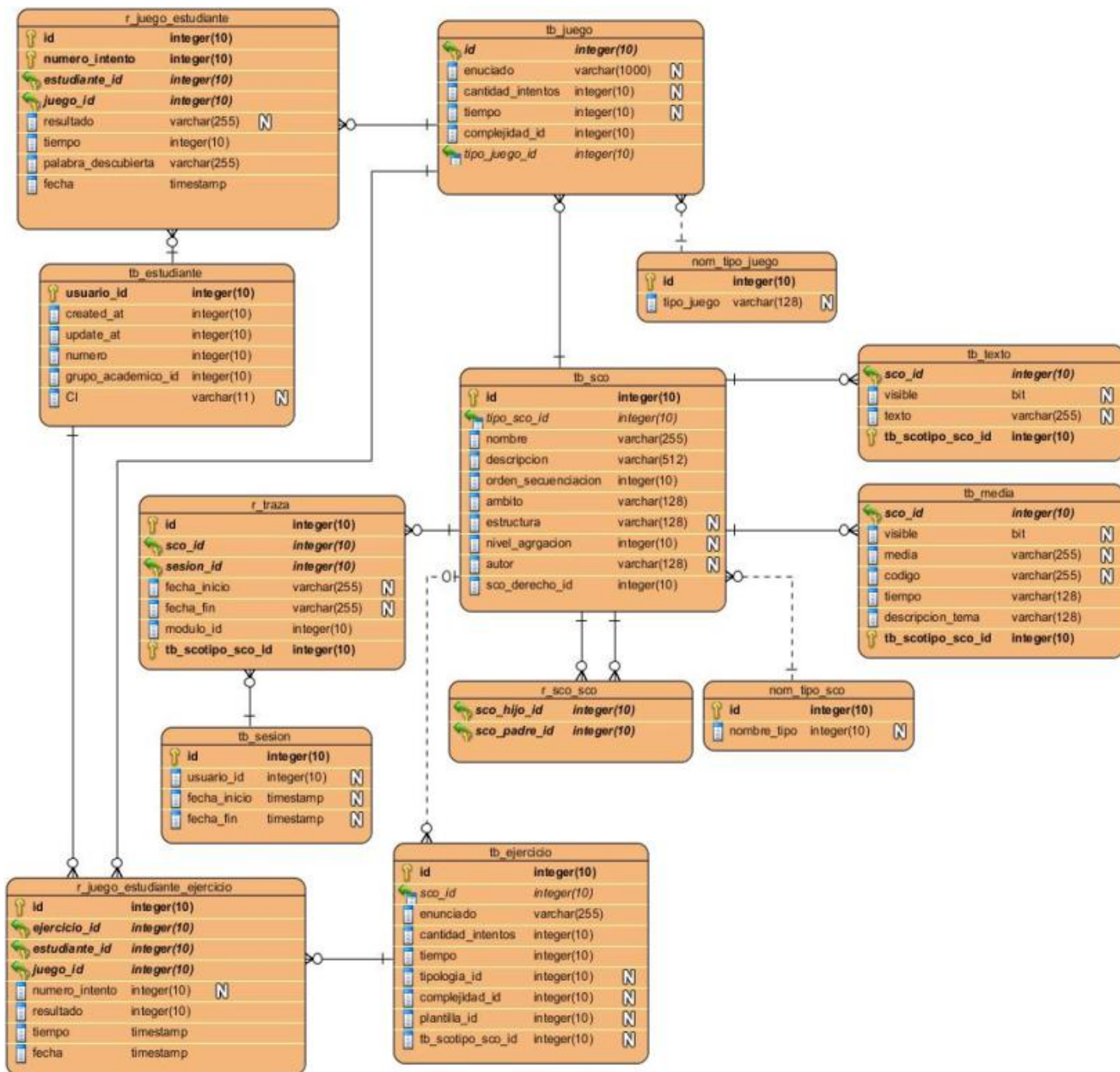


Figura 3.22 Modelo físico de datos.

4.4 Prueba

En el proceso de desarrollo de software se define una etapa de pruebas, con el objetivo de analizar si el producto final fue implementado correctamente, para que la solución se considere lista es necesario tener una vía para validar que lo que se realizó realmente es correcto y cumple con las expectativas del cliente o usuario final. Con las pruebas se verifican que todos los requerimientos hayan sido implementados y se determinan los defectos en el software. Existen dos métodos de pruebas: pruebas de caja blanca, estas pruebas comprueban los caminos lógicos dentro del sistema y son derivadas a partir de las especificaciones internas del diseño o del código. Otro de los métodos son las pruebas de caja negra las cuales pretenden demostrar que las entradas al software se aceptan de forma adecuada y se produce el resultado correcto. (36)

Las pruebas de caja negra se centran fundamentalmente en los requisitos funcionales del software. Estas se realizan sobre la interfaz de la aplicación, con el objetivo de demostrar que las funciones son correctas, que la entrada se acepta de forma adecuada y que se produce una salida correcta. A través de las mismas es muy usual encontrar funciones incorrectas o ausentes, errores de interfaz, errores de rendimiento, así como errores de inicialización o terminación. (37) Dada la necesidad de que las pruebas abarquen los requerimientos, las funciones y las respuestas de la aplicación, se decide utilizar las pruebas de caja negra.

Los casos de prueba son un conjunto de condiciones o variables bajo los cuáles se determina si las funcionalidades de la aplicación son correctas. Estos incluyen una descripción de la funcionalidad que se probará, que es tomada de los requisitos o de los casos de uso. (36) Para consultar como fueron diseñados los casos de pruebas, ver los anexos, sección Anexo 6 Diseño de los casos de pruebas.

Resultado de las pruebas

A partir de los casos de prueba se realizaron tres iteraciones, en las cuales se probó la aplicación con un alto grado de detalle. Las pruebas realizadas arrojaron resultados visibles, estas contribuyeron a mejorar la calidad y funcionalidad del módulo Juegos. A continuación se muestra una tabla con el resumen de las tres iteraciones realizadas:

Clasificaciones	1ra iteración	2da iteración	3ra iteración
-----------------	---------------	---------------	---------------

Alta	2	1	
Media	1	1	
Baja	1	1	

Figura 3.22 Resumen de No conformidades encontradas

Luego de concluida cada iteración de pruebas se analizaron y erradicaron las no conformidades encontradas, en cada una se comprobó que hubiesen sido corregidos los errores detectados en la iteración anterior y se abarcó la verificación del correcto funcionamiento de todos los requisitos del sistema, lo que contribuyó a mejorar la calidad y funcionalidad del software. Luego de la tercera iteración no se encontraron no conformidades por lo que se decidió concluir las pruebas.

4.5 Conclusiones

En este capítulo se representa el sistema a partir de sus componentes de implementación. Para detallar descriptivamente la solución propuesta, se hizo un análisis de la estandarización del código y se realizaron los diagramas de despliegue y componentes, quedando confeccionada la vista estática del sistema. Se realizaron además las pruebas al módulo Juegos, asegurando así el correcto funcionamiento del mismo.

Conclusiones generales

Luego de la realización de este trabajo se puede arribar a las siguientes conclusiones:

- Se alcanzó un mejor entendimiento de las características del sistema, mediante la realización del modelo de dominio, la identificación de los requisitos funcionales y no funcionales así como la elaboración del diagrama de casos de uso del sistema y la descripción detallada de cada uno de estos.
- A través del análisis y diseño se obtuvieron los diagramas de clases que describen cómo implementar el sistema.
- Se realizó la implementación del módulo de acuerdo con las funcionalidades definidas.
- Se llevaron a cabo las pruebas al módulo Juegos, permitiendo erradicar las deficiencias encontradas y asegurando así el correcto funcionamiento del mismo.

Se puede concluir que se logró satisfactoriamente el cumplimiento de los objetivos propuestos para el trabajo de diploma. La investigación que se llevó a cabo, junto a los conocimientos adquiridos durante el proceso de desarrollo, hicieron posible realizar el módulo Juegos de la colección El Navegante en su versión multiplataforma, haciendo uso de las herramientas y tecnologías adecuadas.

Recomendaciones

A partir del trabajo realizado se recomienda al proyecto Multisaber-El Navegante:

- Continuar incorporando otros juegos interactivos y educativos al módulo Juegos de la colección El Navegante.
- Implementar una sala de juegos que brinde la posibilidad de que dos usuarios puedan jugar en línea el juego seleccionado.

Referencias bibliográficas

1. **Ramos Pérez L, Domínguez Lovaina J, Gavilondo Mariño X, Fresno Chávez C.** *¿Software educativo, hipermedia o entorno educativo?* Ciudad de La Habana, Cuba: Centro Nacional de Información de Ciencias Médicas-Infomed, 2008. Disponible en: http://bvs.sld.cu/revistas/aci/vol18_4_08/aci61008.htm.
2. **Núñez Rojas, Nemecio.** Concepción del Software educativo. *La Webquest, el aula virtual y el desarrollo de competencias para la investigación en los estudiantes del I Ciclo de educación – USAT.* 2010. Disponible en: www.eumed.net/libros/2010a/669/.
3. Coordinación General de Investigación y Desarrollo de Modelos Educativos. [En línea] [Citado el: 10 de enero de 2012.] http://investigacion.ilce.edu.mx/panel_control/doc/c36,evaluacsoft.pdf.
4. **Marquès, Pere.** El software educativo. [En línea] Universidad Autónoma de Barcelona. [Citado el: 10 de enero de 2012.] http://www.lmi.ub.es/te/any96/marques_software/.
5. **Galvis Panqueva, Álvaro.** SlideShare Inc. [En línea] [Citado el: 10 de enero de 2012.]
6. **Rosario, Jimmy.** *La Tecnología de la Información y la Comunicación (TIC). Su uso como Herramienta para el Fortalecimiento y el Desarrollo de la Educación Virtual.* 2005. Disponible en el ARCHIVO del Observatorio para la CiberSociedad en <http://www.cibersociedad.net/archivo/articulo.php?art=218>.
7. **Graells, Pere Marquès.** MULTIMEDIA EDUCATIVO: CLASIFICACIÓN, FUNCIONES, VENTAJAS, DISEÑO DE ACTIVIDADES. [En línea] 3 de agosto de 2010. [Citado el: 10 de enero de 2012.] <http://peremarques.pangea.org/funcion.htm>.
8. **Lamarca Lapuente, María Jesús.** *Hipertexto: El nuevo concepto de documento en la cultura de la imagen.* [En línea] [Citado el: 11 de enero de 2012.] <http://www.hipertexto.info/documentos/hipermedia.htm>.
9. **MsC. Rodríguez, Mario del Toro.** Modelos de diseño didáctico de hiperentornos de enseñanza-aprendizaje. *Resumen de la Tesis en opción al grado de Doctor en Ciencias Pedagógicas.* Ciudad de La Habana, República de Cuba: Instituto Superior Pedagógico "Enrique José Varona", 2006.
10. **M. Sc. Orestes Coloma Rodríguez.** [En línea] [Citado el: 11 de enero de 2012.] <http://www.forumcyt.cu/UserFiles/forum/Textos/1101687.pdf>.
11. *Desarrollo del Módulo General de la colección El Navegante. Resumen de la Tesis en opción al título en Ciencias Informáticas. Ciudad de La Habana, República de Cuba: Universidad de las Ciencias Informáticas (UCI), 2011.*
12. **GALLEGO, JUAN PABLO GOMEZ.** Fundamentos de la Metodología RUP. Universidad Tecnológica de Pereira.2007.
13. **Jesús, Areba Barranco.** Metodología del análisis estructurado de sistemas.2001.
14. Servidores Web. *Masadelante.com.* [En línea] 2010. [Citado: 12 de enero de 2012.] <http://www.masadelante.com/faqs/servidor>.
15. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James.** . *El Proceso Unificado de Desarrollo, La guía completa del Proceso Unificado, escrita por sus creadores.* Madrid : Addison Wesley, 2000.
16. Applying UML and Patterns -An Introduction to Object- Oriented Analysis and Design and Iterative Development. **Larman, Craig.** 2005, Vol. tercera edición.
17. **Landeros Gómez, Ruth Priscila y Del Juncal Huerta, Jorge Luis.** Herramientas Case. s.l. : Universidad Veracruzana, 2007.
18. **Damián Pérez Valdés .** Maestros del web. Los diferentes lenguajes de programación para la web. [En línea] [Citado el: 12 de enero de 2012.] <http://www.maestrosdelweb.com/principiantes/los-diferentes-lenguajes-de-programacion-para-la-web/>.
19. librosweb.es. [En línea] [Citado el: 12 de enero de 2012.] <http://www.librosweb.es/javascript/capitulo1.html>.
20. Los diferentes lenguajes de programación para la web. New Web Star. [En línea] [Citado el: 16 de Diciembre de 2010.] <http://www.newwebstar.com/ebooks/133193-los-diferentes-lenguajes-de-programaciun-para-la.html>.
21. maestros del web. [En línea] [Citado el: 14 de enero de 2012.] <http://www.maestrosdelweb.com/editorial/javascript-facil-y-rapido-con-jquery/>.
22. **Sánchez Suárez, Jose Manuel.** <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=jQueryIntro>. [En línea] [Citado el: 14 de enero de 2012.]
23. **Potencier, F. y Zaninotto, F.** LibrosWeb.es. [En línea] [Citado el: 14 de enero de 2012.] http://librosweb.es/symfony_1_2/capitulo1/symfony_en_pocas_palabras.html.
24. librosweb.es. [En línea] [Citado el: 14 de enero de 2012.] http://www.librosweb.es/symfony_1_2/.

25. IBM WebSphere Commerce. Guías de programación y aprendizaje. [En línea] [Citado el: 15 de enero de 2012.] <ftp://ftp.software.ibm.com/software/websphere/commerce/55/es/WC55ProgrammingGuide.pdf>.
26. [En línea] [Citado el: 15 de enero de 2012.] <http://petra.euitio.uniovi.es/~i1667065/HD/documentos/Entornos%20de%20Desarrollo%20Integrado.pdf>.
27. [En línea] <http://netbeans.org>.
28. Ciberneta. Conceptos básicos de servidor Web. [En línea] [Citado: 17 de enero de 2012.] http://www.ciberneta.com/manuales/instalacion_servidor_web/1_conceptos_basicos.php.
29. ciberaula. [En línea] http://linux.ciberaula.com/articulo/linux_apache_intro/.
30. Computer Audio Video Systems Integrator. CAVSI. [En línea] [Citado el: 17 de enero de 2012.] <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgbd/>.
31. PostgreSQL. [En línea] <http://www.postgresql.org/about/>.
32. **Joaquin Gracia**. IngenieroSoftware. [En línea] [Citado el: 10 de febrero de 2012.] <http://www.ingenierosoftware.com/analisisydiseno/patrones-diseno.php>.
33. **Lic Espinoza, Robles**. El flujo de análisis. [En línea] [Citado el: 15 de Febrero de 2012.] <http://www.slideshare.net/juliopari/13-clase-flujo-de-analisis>.
34. **Martínez, Alejandro y Martínez, Raúl**. Proceso Unificado de Desarrollo. Escuela Politécnica Superior de Albacete – Universidad de Castilla la Mancha. [En línea] <http://www.dsi.uclm.es/asignaturas/42551/trabajosAnteriores/Trabajo-Guia%20RUP.pdf>
35. Marca, H. y Quisbert, N. Diagrama de Despliegue, [Citado el: 18 de febrero de 2012]. Disponible en: <http://virtual.usalesiana.edu.bo/web/practica/archiv/despliegue.doc>.
36. **Binder, Robert**. Object-Oriented Software Testing. Communications of the ACM. September, 1994.
37. **Pressman, Roger S**. Ingeniería del Software, un enfoque práctico.
38. **COLOMA Rodríguez, Orestes**. Hiperentorno de aprendizaje “EUREKA”: un software educativo para la enseñanza de la matemática. Holguín, Cuba: Instituto Superior Pedagógico “José de la Luz y Caballero”. [Citado el: 30 de mayo de 2012.]
39. **PÉREZ de Villa Ail Sellés** [et al]. Software para la educación ambiental de los estudiantes de secundaria. Revista electrónica de la Agencia de Medio Ambiente. 2009, no. 17. [Citado el: 30 de mayo de 2012].
40. **LATORRE, Guillermo**. GCompris: software libre con actividades y juegos educativos para niños. 24 de marzo de 2008. [Citado el: 30 de mayo de 2012.] Disponible en: <http://www.genbeta.com/herramientas/gcompris-software-libre-con-actividades-y-juegos-educativos-para-ninos>.
41. **SANDRA Quero, Ramones, MADUEÑO Madueño, Leonel**. Süchiki Walekerü: Un ejemplo del uso de las TIC en escuelas indígenas. Caso Wayuu. Zulia, Venezuela: Universidad de Zulia. 2006. [Citado el: 30 de mayo de 2012.]
42. Questionmark. [Citado el: 30 de mayo de 2012.] Disponible en: <http://www.questionmark.com/esp/index.aspx>
43. Crear ejercicios virtuales con EDILIM. Universidad de Salamanca. [Citado el: 30 de mayo de 2012.] Disponible en: <http://web.usal.es/~angelpoveda/web%20biologia/tutoriales/EDILIM/index.html>.