

Universidad de Sancti Spíritus “José Martí Pérez”

Facultad de Ingeniería

Carrera Ingeniería Informática



Título: “Aplicación Web para contribuir al proceso de enseñanza-aprendizaje de la Ciencia Química de onceno grado en Cuba.”

Trabajo de diploma para optar por el título de Ingeniería Informática.

Autor: Pedro Alexis López Valdivia

Tutor: Ernesto Rodríguez López

Consultante: Alexis López Clemente

Pablo Manuel Sánchez Portal

Sancti Spíritus

2014

Pensamiento

Exígete mucho a ti mismo y espera poco de los demás. Así te ahorrarás disgustos.

Confucio.

Agradecimientos

La Familia:

- *A mi mamá por ser la mejor madre y persona más maravillosa del mundo.*
- *A mi abuela Oneida por estar siempre presente.*
- *A mi padre por brindarme siempre apoyo y confianza.*
- *A mis dos hermanos Elizabeth y César por darme la alegría de su presencia en mi vida.*
- *A mis tías por darme mucho amor y cariño.*
- *A mis tíos por estar cuando los necesito incondicionalmente.*
- *A mis primos por los buenos momentos que hemos pasado juntos, y por los que faltan por pasar que no son pocos.*

Particulares:

- *A mis amigos que me han apoyado en estos años de estudio.*
- *A mi tutor Ernesto por tener la paciencia y constancia en momentos de desesperación.*
- *A Pablo por brindarme su tiempo y conocimiento aun cuando él lo necesitara para sus proyectos.*

Dedicatoria

A mi mamá y mi papá, pues su mayor deseo es que yo me convierta en un profesional.

A todo aquel que de una manera u otra puso su mano para que yo llegara hasta aquí.

Resumen

Debido a la insuficiencia de materiales informatizados en la asignatura de Química en onceno grado se decidió llevar a cabo un proyecto con el objetivo de solucionar en gran medida el problema en cuestión, dando a los profesores y estudiantes una herramienta con las condiciones necesarias para el estudio y el aprendizaje de la materia en la enseñanza preuniversitaria en el país. En la elaboración del sistema se llevaron a cabo varias etapas las que fueron planteadas por el Proceso Unificado de Desarrollo (RUP), utilizando como herramienta de análisis y diseño: Visual Paradigm for UML Enterprise Edition. Para la implementación del sistema se utilizó como lenguaje de programación PHP para obtener el máximo rendimiento de y aprovechar todas sus características; para la persistencia de los datos se utilizó PostGreSQL server y como servidor Web se utilizó el WampServer.

Abstract

Due to insufficient digitized materials in the subject of Chemistry in eleventh grade was decided to carry out a project with the aim of solving the problem greatly concerned, giving teachers and students a tool with the necessary conditions for the study and learning of the subject in pre-university education in the country. In developing the system were carried out several stages which were raised by the Rational Unified Process (RUP), using analysis and design tool: Visual Paradigm for UML Enterprise Edition. To implement the system was used as the programming language PHP to get the most out of and take advantage of all its features, to the persistence of the data was used as PostgreSQL server and Web server was used WampServer.

Tabla de contenido

Introducción.....	1
Capítulo 1: Fundamentación teórica y metodológica que sustenta la elaboración de una Aplicación Web para apoyar el proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura del onceno grado en la enseñanza preuniversitaria en Cuba.....	6
1.1 El proceso de Enseñanza-Aprendizaje	6
1.2 Medios de enseñanza.....	7
1.2.1 Las Tecnologías de la Información y las Comunicaciones (TIC).....	8
1.2.2 Las TIC empleadas a la Educación	8
1.3 La computadora como medio de enseñanza.....	9
1.3.1 Software Educativo	10
1.3.2 Sitios web utilizados para la gestión de información.....	10
1.3.3 Sitio web, tecnologías web utilizadas para su desarrollo Sitio web.....	11
1.4 Tecnologías Web	12
1.4.1 Arquitectura cliente-servidor con tres capas	13
1.5 Patrón MVC	15_Toc389206604
1.6 Tecnologías del lado del cliente.....	16
1.7 Tecnologías del lado del servidor	19
1.8 Servidores web.....	20
1.9 Sistema Gestores de Base de Datos	21
1.10 Lenguaje de Modelado Unificado (UML)	24
1.10.1 Proceso Unificado de Desarrollo (RUP).....	25
1.11 Herramientas de desarrollo.....	26
1.11.1 El editor de texto CKEditor	26
1.11.2 Macromedia Dreamweaver	26
Capítulo II: Diseño de una Aplicación Web que contribuya a mejorar el aprendizaje de la asignatura Química en onceno grado.....	29
2.1 Modelo del negocio.....	29
2.1.2 Reglas del negocio a considerar.....	29
2.1.3 Modelado del negocio.....	30
2.1.4 Actores del Negocio.....	30
2.1.5 Trabajadores del Negocio	31
2.1.6 Diagrama de Casos de Uso del Negocio.....	31
2.1.7 Descripción de los Casos de Uso del negocio.....	31
2.1.8 Diagrama de actividades	32
2.1.9 Modelo de objetos del negocio	33
2.2 Especificación de los requisitos de software	33
2.2.1 Requerimientos funcionales	33
2.2.2 Requerimientos no funcionales	36
2.3- Descripción del sistema propuesto	38
2.3.1 Modelo de casos de uso del sistema.....	38
2.3.2 Actores del sistema	38
2.3.3 Casos de uso del sistema.....	39
2.3.4 Diagramas de casos de uso del sistema.....	39
2.3.5 Descripción de los casos de uso del sistema	42
Conclusiones Parciales del capítulo.....	50
Capítulo 3: Implementación de la aplicación propuesta para apoyar el proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura de onceno grado en la enseñanza preuniversitaria en Cuba:.....	51
Introducción	51

3.1 Diagrama de Clases Web	51
3.2 Diseño de la base de datos	61
3.2.1 Diagrama de clases persistentes	62
3.2.2 Diagrama del Modelo Físico de Datos	62
3.3 Principios de diseño	63
3.3.1 Estándares en la interfaz de la aplicación	64
3.3.2 Concepción general de la ayuda.....	64
3.3.3 Tratamiento de errores	64
3.3.4 Seguridad	65
3.4 Modelo de implementación.....	65
3.4.1 Diagrama de despliegue	65
3.4.2 Diagrama de componentes	66
Conclusiones parciales del capítulo	66
Conclusiones:	67
Recomendaciones	68
Bibliografía.....	69
Anexos	71

Tablas

Tabla 1: Actor del negocio.....	30
Tabla 2 Trabajador del negocio.....	31
Tabla 3 Descripción del Caso de Uso del Negocio.....	31
Tabla 4: Actores del sistema.....	39
Tabla 5. Descripción del caso de uso Crear cuenta.....	42
Tabla 6. Descripción del caso de uso Autenticar usuario.....	42
Tabla 7. Descripción del caso de uso Cambiar contraseña.....	43
Tabla 8. Descripción del caso de uso Gestionar usuario.....	43
Tabla 9. Descripción del caso de uso Gestionar Asignatura.....	44
Tabla 10. Descripción del caso de uso Gestionar Estudiantes.....	44
Tabla 11. Descripción del caso de uso Gestionar Estudiantes.....	44
Tabla 12. Descripción del caso de uso Gestionar programas.....	45
Tabla 13. Descripción del caso de uso Gestionar unidad.....	45
Tabla 14. Descripción del caso de uso Gestionar epígrafe.....	46
Tabla 15. Descripción del caso de uso Gestionar bibliografía.....	46
Tabla 16. Descripción del caso de uso Gestionar glosario.....	47
Tabla 17. Descripción del caso de uso Gestionar ejercicios.....	47
Tabla 18. Descripción del caso de uso Gestionar entidad.....	48
Tabla 19. Descripción del caso de uso Gestionar grupo.....	48
Tabla 20. Descripción del caso de uso Gestionar calendario.....	50
Tabla 21. Descripción del caso de uso Gestionar anuncios.....	49
Tabla 22. Descripción del caso de uso Gestionar noticias.....	49
Tabla 23. Descripción del caso de uso Gestionar evento.....	50

Introducción

No se puede entender el mundo de hoy sin un mínimo de cultura informática. Es preciso entender cómo se genera, cómo se almacena, cómo se transforma, cómo se transmite y cómo se accede a la información en sus múltiples manifestaciones (textos, imágenes, sonidos) si no se quiere estar al margen de las corrientes culturales (Eugenia, 2005).

Estas tecnologías han llegado a ser uno de los pilares básicos de la sociedad y hoy es necesario proporcionar al ciudadano una educación que tenga en cuenta esta realidad. Las posibilidades educativas de las TIC (Tecnologías de la Informática y las Comunicaciones) han de ser consideradas en dos aspectos: su conocimiento y su uso. La relación entre las TIC y la educación tiene dos vertientes: por un lado, los ciudadanos se ven abocados a conocer y aprender sobre las TIC; por otro, las TIC pueden aplicarse al proceso educativo. Este doble aspecto se refleja en dos expectativas educativas distintas: por un lado, tenemos a los informáticos, interesados en aprender informática, y en el otro, a los profesores, interesados en el uso de la informática para la educación (Rosario, 2005).

La educación en Cuba está sujeta a procesos de cambios renovadores en la actualidad donde juega un papel preponderante el uso constante de las TIC, ya que las mismas son una herramienta importante pues brindan una gama infinita de aplicaciones, las cuales pueden ser usadas en la informatización de la educación en nuestro país, lo que contribuye a optimizar el proceso docente educativo, donde los estudiantes y profesores logren acceder a un proceso de aprendizaje más ameno y eficaz, donde cada uno de manera individual interactúe con estas novedosas tecnologías.

Es vital que la escuela cubana acerque a los estudiantes a la cultura de hoy. Por ello es importante la presencia en clases del ordenador, la cámara de vídeo y de la televisión, como medios de enseñanza utilizados con fines diversos como son actividades lúdicas, informativas, comunicativas e instructivas. Por lo que nos queda claro que la educación no puede quedar al margen de estos cambios. Independientemente de la función que tiene el trabajo metodológico en el perfeccionamiento del proceso de enseñanza aprendizaje al proveer al profesor de herramientas y métodos para la impartición de las clases, es una fuente de

actualización y aprendizaje para los docentes. Muchos autores coinciden en afirmar que a través de los planes de trabajo metodológico en los diferentes niveles organizativos concebidos en la universidad, el docente renueva constantemente sus saberes en el campo de la didáctica y la pedagogía (Cruz Rodríguez, 2010; Horruitiner Silva, 2005; Sobrino Pontigo, 2012). Pues del mismo modo que en su día sucedió con la introducción de la escritura o la aparición de la imprenta, el auge de las TIC ha transformada nuestro modo de pensar, de relacionarnos e incluso la manera de enseñar y de aprender (Graells, 2011).

Tal es el caso de la enseñanza preuniversitaria en Cuba que presenta algún grado de informatización en cuanto a algunas asignaturas como son la Biología, la Geografía y la Historia, pero existen grandes carencias en cuanto a la informatización de otras asignaturas como son la Matemática, la Física y la Química, especialidad que abordaremos en nuestra investigación en la que se pudo verificar en este nivel, la informatización existe a nivel de algunas tele clases, que se transmiten por las frecuencias de nuestros canales educativos una vez a la semana, en ellas se imparten algunos contenidos que deben ser recibidos por los estudiantes en la semana en curso, lo cual contribuye en algo a saldar la brecha existente entre las diferentes localidades del país que algunas carecen de claustros con suficiente especialización y experiencia pedagógica para lograr la excelencia del proceso de enseñanza aprendizaje en esta asignatura en la enseñanza media superior, lo que garantiza un equilibrio parcial a nivel de país.

Se ha constatado que existen grandes carencias de materiales informatizados en relación a la Química como asignatura en la enseñanza preuniversitaria y los existentes se encuentran muy dispersos. Debido a la complejidad, diversidad y profundidad de los contenidos impartidos de forma curricular, es el oncenno grado el de mayor dificultad para lograr las habilidades cognitivas y procedimentales que deben adquirir estos estudiantes, lo que motiva a escoger este grado para realizar el proceso de informatización de esta asignatura para con ello disminuir la brecha existente entre las diferentes regiones del país como se expresó anteriormente.

En estudios realizados se ha podido verificar que se carece de bases de datos donde se garantice la informatización de los contenidos pertenecientes a la asignatura de Química en el oncenno grado, por lo que se dificulta la presentación de estos de forma amena e interactiva, así mismo se constató la inexistencia de

materiales de apoyo a la docencia en forma digital o impresa que permitan la profundización del contenido por los estudiantes y profesores de esta asignatura, por lo que el diseño de un software al respecto genera niveles superiores en la adquisición de conocimiento en relación a la asignatura antes mencionada por parte de estudiantes y profesores.

Teniendo en cuenta que la emisión de un programa televisivo no es suficiente para la asimilación del contenido y desarrollo de las habilidades necesarias para el correcto dominio de esta asignatura ya que las transmisiones pueden ser interrumpidas por diferentes variables físicas, meteorológicas y ajenas al proceso de enseñanza aprendizaje de esta asignatura, lo que presupone plantear el siguiente **problema científico**.

¿Cómo contribuir al proceso de enseñanza-aprendizaje de la Ciencia Química como asignatura del octavo grado de la Enseñanza Preuniversitaria en Cuba?

Para solucionar el problema científico planteado al inicio de la investigación nos planteamos el siguiente **objetivo general**:

Desarrollar una Aplicación Web para contribuir al proceso de enseñanza-aprendizaje de la Ciencia Química como asignatura del octavo grado en la Enseñanza Preuniversitaria en Cuba.

Para darle cumplimiento al objetivo surgieron las siguientes **preguntas de investigación**:

1. ¿Qué fundamentos teóricos y metodológicos sustentan la elaboración de una herramienta informática que contribuya al proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura?
2. ¿Cómo diseñar una Aplicación Web para contribuir al proceso enseñanza - aprendizaje de la Ciencia Química como asignatura del octavo grado en la Enseñanza Preuniversitaria en Cuba?
3. ¿Cómo implementar una Aplicación Web para contribuir al proceso enseñanza - aprendizaje de la Ciencia Química como asignatura del octavo grado en la Enseñanza Preuniversitaria en Cuba?

Para dar respuesta a las preguntas de investigación se trazaron las siguientes **tareas de investigación**:

1. Determinar los fundamentos teóricos y metodológicos que sustentan la elaboración de una herramienta informática que contribuya al proceso enseñanza - aprendizaje de la Ciencia Química como asignatura.
2. Diseñar una Aplicación Web para contribuir al proceso enseñanza - aprendizaje de la Ciencia Química como asignatura del octavo grado en la Enseñanza Preuniversitaria en Cuba.
3. Implementar una Aplicación Web para contribuir al proceso enseñanza - aprendizaje de la Ciencia Química como asignatura del octavo grado en la enseñanza preuniversitaria en Cuba.

El trabajo está conformado por: introducción, tres capítulos, conclusiones, recomendaciones y bibliografía consultada. A continuación se presenta una breve descripción de cada capítulo:

Capítulo 1. Fundamentación teórica y metodológicos que sustentan la elaboración de una herramienta informática que apoye el proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura del octavo grado en la enseñanza preuniversitaria en Cuba. En este capítulo se abordan la fundamentación teórica del tema y los conceptos asociados al negocio del problema. Se explica en detalles el problema a resolver y se describe las tecnologías y metodologías utilizadas para su desarrollo teniendo en cuenta las tendencias actuales en el desarrollo de las TIC.

Capítulo 2. Descripción de la solución propuesta para apoyar el proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura del octavo grado en la enseñanza preuniversitaria en Cuba. Este capítulo se centra fundamentalmente en analizar el modelo del negocio, así como la descripción de dicho proceso utilizando los artefactos de UML. Además se muestran los diagramas y modelos de casos de uso utilizados en el sistema a construir, con su correspondiente descripción, así como los requisitos funcionales y no funcionales.

Capítulo 3. Implementación de la aplicación propuesta para apoyar el proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura

del onceno grado en la enseñanza preuniversitaria en Cuba. En este capítulo se describe el diseño de la solución propuesta con sus correspondientes modelos y/o diagramas: modelo de clases del diseño, modelo de clases persistentes, modelo de despliegue y el modelo de datos. Los cuales ayudarán al entendimiento del funcionamiento de la Aplicación Web.

Capítulo 1: Fundamentación teórica y metodológica que sustenta la elaboración de una Aplicación Web para apoyar el proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura del oncenso grado en la enseñanza preuniversitaria en Cuba

1.1 El proceso de Enseñanza-Aprendizaje

Enseñanza y aprendizaje está basado en el proceso de relación que se establece entre el estudiante y el profesor con el fin de lograr una formación integral del alumno, donde a través de la retroalimentación entre ambos entes se consolida, se fijan y se aplican los conocimientos ya presentes en los estudiantes y se muestra lo desconocido.

Esto implica que hay un sujeto que conoce (el que puede enseñar), y otro que desconoce (el que puede aprender). El que puede enseñar, quiere y sabe enseñar (el profesor); El que puede aprender quiere y sabe aprender (el alumno). Ha de existir una disposición por parte de alumno y profesor. Aparte de estos agentes, están los contenidos, esto es, lo que se quiere enseñar o aprender (elementos curriculares) y los procedimientos o instrumentos para enseñarlos o aprenderlos (medios). Cuando se enseña algo es para conseguir alguna meta (objetivos). Por otro lado, el acto de enseñar y aprender acontece en un marco determinado por ciertas condiciones físicas, sociales y culturales (contexto) (Castellano, 2002).

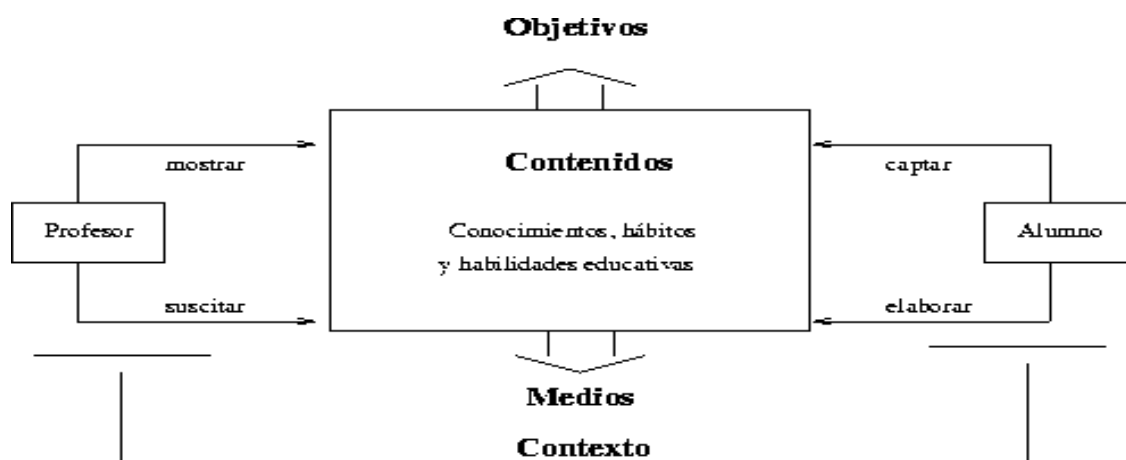


Figura 1: Elementos del proceso Enseñanza-Aprendizaje.

El proceso de enseñar es el acto mediante el cual el profesor muestra o suscita contenidos educativos (conocimientos, hábitos, habilidades) a un alumno, a través de unos medios, en función de unos objetivos y dentro de un contexto. El proceso de aprender es el proceso complementario de enseñar. Aprender es el acto por el cual un alumno intenta captar y elaborar los contenidos expuestos por el profesor, o por cualquier otra fuente de información. Él lo alcanza a través de unos medios (técnicas de estudio o de trabajo intelectual). Este proceso de aprendizaje es realizado en función de unos objetivos, que pueden o no identificarse con los del profesor y se lleva a cabo dentro de un determinado contexto (Hernández, 1989).

El proceso de Enseñanza-Aprendizaje debe considerarse como un sistema estrechamente vinculado con la actividad práctica del hombre, que en última instancia, condiciona sus posibilidades de conocer, comprender y transformar la realidad objetiva.

1.2 Medios de enseñanza

Se considera medio de enseñanza a todos los componentes del proceso docente que actúan como soporte material de los métodos (instructivos o educativos), con el propósito de lograr los objetivos planteados (Campos, 2009).

La escuela, como institución integrada y parte constitutiva del entramado social, no es ajena al desarrollo tecnológico. Aunque siempre ha estado rezagada en la introducción de los nuevos medios en relación a la evolución del resto de la sociedad, inevitablemente los procesos de enseñanza tienden cada vez a emplear distintos tipos de tecnologías. La propia sociedad así lo demanda (Castillo, 2008).

El concepto de medios de enseñanza y el papel que juega dentro del proceso docente educativo se ha definido por pedagogos de todas partes del mundo. En un principio fueron denominados “auxiliares” que empleaban los maestros para sus clases, lo cual era algo erróneo ya que son componentes del proceso sistémico del cual no se puede separar. Se pudiera definir que los medios de enseñanza son diferentes imágenes y representaciones de objetos y fenómenos, que se crean para la docencia. También objetos naturales e industriales, que

contienen información y se emplean como fuente del conocimiento (Campos, 2009).

1.2.1 Las Tecnologías de la Información y las Comunicaciones (TIC)

Se puede denominar como Tecnologías de la Información y la Comunicación (TIC) el conjunto de procesos y productos derivados de las nuevas herramientas (hardware y software), soportes de la información y canales de comunicación relacionados con el almacenamiento, procesamiento y transmisión digitalizados de la información (Romero, 2004).

La utilización de tecnología, nuevas tecnologías, nuevas tecnologías de la información y la comunicación o, simplemente, tecnologías de la información y la comunicación se ha hecho de forma indistinta, y centrándose más en el espacio temporal que en las características de los instrumentos utilizados, para referirse a concepciones o ámbitos de estudio similares (Álvarez & Mayo, 2009).

Las TIC son una realización social que facilitan los procesos de información y comunicación, gracias a los diversos desarrollos tecnológicos, en aras de una construcción y extensión del conocimiento que derive en la satisfacción de las necesidades de los integrantes de una determinada organización social (Álvarez & Mayo, 2009).

Las definiciones ofrecidas de las nuevas tecnologías de la información y comunicación son diversas, variadas y algunas veces incluso contradictorias, entre otros motivos por el simple hecho de que el propio término "nuevo" ya implica caducidad. Aunque suele existir un acuerdo en considerarlas como aquellos instrumentos técnicos que giran en torno a la información y a la comunicación, y a los nuevos descubrimientos que sobre las mismas se van originando (Borges, 2007).

1.2.2 Las TIC empleadas a la Educación

A finales del siglo pasado, la casi totalidad de los campos profesionales han visto incrementado su potencial de desarrollo con la incorporación de las nuevas tecnologías y ello ha motivado un cambio sustancial en el modo de ejercer las funciones específicas en cada uno de estos campos. Esta situación lleva como contrapartida la demanda implícita de nuevos roles para el desarrollo profesional. En el caso concreto de la educación no hay excepción, a lo largo de estos últimos

años, se enfatiza mucho en relación con el cambio en el perfil del maestro como consecuencia de la integración de las nuevas tecnologías en el ámbito escolar (Conociendo algo de software educativo, 2009).

Las TIC son empleadas en los diversos medios de transmisión de la información, insertándose así en diferentes esferas de la vida cotidiana, y entre ellas, en el proceso docente – educativo. En las condiciones de hoy en día es posible disponer de herramientas de apoyo al proceso educativo que se hallen más cercanas a la manera de percibir y entender el mundo, tanto por parte de los niños, como de los jóvenes y adultos, de una manera dinámica, llena de estímulos paralelos, capacitados para el cambio constante e integrado (Martínez, 2009).

Las TIC han incidido en todos los campos relacionados con la educación, facilitando la transformación y la optimización de la mayoría de los procesos administrativos, el desarrollo de metodologías innovadoras en los procesos de enseñanza-aprendizaje, el acceso a la formación superior de nuevos grupos de personas, así como una incipiente transformación del sistema organizativo de la sociedad (Conociendo algo de software educativo, 2009).

Las nuevas exigencias en la educación se centran en la mejora del proceso educativo y, en este sentido, la integración de las TIC facilita aspectos relacionados con la mejora del trabajo individual, la autonomía del alumnado, la facilidad para el desarrollo de trabajos en equipo y colaborativos, la posibilidad de modificar y adaptar los métodos de evaluación y la interacción bidireccional entre el profesorado y el alumnado.

1.3 La computadora como medio de enseñanza

El desarrollo de la ciencia y la técnica trajo asociado el uso de la computadora como medio de enseñanza ya que la misma constituía una herramienta poderosa para la enseñanza y el aprendizaje dada las cualidades que poseía. Este medio permite que aquellas personas interesadas en aprender contaran con recursos como son imágenes, videos, libros, explicaciones habladas, así como programas encaminados a la educación. Es conocido y constituye un hecho para el mundo que una vez empleada la computadora como medio de enseñanza se lograron alcanzar mejores resultados en la educación (Campos, 2009).

Una computadora con un buen software puede producir un aprendizaje con calidad con respecto a ciertos procedimientos de trabajo, pero también un mal uso

de este puede traer aparejadas dificultades de aprendizaje muy serias para el estudiante. En ocasiones hemos encontrado personas que se preguntan: ¿el uso de los medios y en particular de las computadoras originará automáticamente un aprendizaje efectivo?. Sencillamente la respuesta sin dudar es que NO. En este sentido, la forma en que se utilice, la motivación a proporcionar, el ajuste de las necesidades educativas, entre otras cuestiones, es lo que propiciará un mejor aprendizaje (Conociendo algo de software educativo, 2009).

1.3.1 Software Educativo

Los software educativos son todos aquellos programas para ordenador, creados con la finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y aprendizaje (Marquez, 2010).

No obstante según esta definición, más basada en un criterio de finalidad que de funcionalidad, se excluyen del software educativo todos los programas de uso general en el mundo empresarial que también se utilizan en los centros educativos con funciones didácticas o instrumentales como por ejemplo: procesadores de textos, gestores de bases de datos, hojas de cálculo, editores gráficos... Estos programas, aunque puedan desarrollar una función didáctica, no han estado elaborados específicamente con esta finalidad (Marquez, 2010).

Para el desarrollo del software educativo (SE), es necesario involucrar a maestros y profesores de cada educación con experiencia suficiente, en el diseño pedagógico y evaluación del proyecto ya implementado, para garantizar el cumplimiento de los requisitos, específicos educativos, planteados para cada software (Catoní, 2009).

1.3.2 Sitios web utilizados para la gestión de información

En el mundo actual los Sitios Web son una de las herramientas más usadas para la gestión de información, debido a que pueden manejar relativamente fácil un gran cúmulo de datos, soportar las consultas hechas por miles de usuarios a la vez, siendo también la confiabilidad y la seguridad una de sus mejores aliadas, conquistando así un sin número de clientes por todo el mundo.

Dentro de las disímiles ramas que han adoptado Sitios Web como vía de desarrollo se encuentra la educación y la enseñanza. Empleando al máximo las

ventajas que brinda un software de este tipo, para la distribución de la información en su beneficio.

1.3.3 Sitio web, tecnologías web utilizadas para su desarrollo Sitio web

Es un conjunto de archivos electrónicos y páginas web referentes a un tema en particular, que incluye una página inicial de bienvenida, generalmente denominada home page, con un nombre de dominio y dirección en Internet específicos (copyright © Informática Milenium, 2009). Algunos de los géneros que destacan son los de negocios, servicios, comercio electrónico en línea, imagen corporativa, entretenimiento y sitios informativos.

Los sitios web pueden ser estáticos o dinámicos según (Escanaverino, 2007):

sitio web estático es el que tiene contenido que no se modifica o actualiza frecuentemente, se mantiene manualmente por alguna persona o personas que usan algún tipo de programa editor con posibilidades para ello.

Sitio web dinámico es el que puede tener cambios o actualizaciones frecuentes en la información que contiene. Cuando el servidor web recibe una petición para una determinada página, la página se genera automáticamente por el software como respuesta directa a la petición; por lo tanto ofrece muchas posibilidades incluyendo por ejemplo: mostrar el estado actual de un diálogo entre usuarios, monitorear una situación cambiante, proporcionar información de alguna manera personalizada a los requisitos del usuario, contestar y reenviar formularios, descargar información desde un FTP (Protocolo de transferencia de ficheros), mantener foros de debate de algún tema, ejecutar búsqueda y localización de información, permitir la publicación de información, facilitar la realización de aprendizajes, realizar gestiones administrativas y comerciales, actuar como medio publicitario, entretener y motivar.

Dentro de las principales ventajas que menciona (Acosta, 2007), están: Facilitar la búsqueda y la localización de información de cualquier tipo y sobre cualquier temática. Suministrar la obtención (o la distribución) de materiales educativos on-line. Proporcionar (o editar) información relacionada con las actividades de los centros e instituciones educativas. Posibilitar la comunicación con otras personas. Publicar en Internet, alcanzando una mayor extensión de la misma. Y preparar la realización de aprendizajes, entre otras.

1.4 Tecnologías Web

Las tecnologías web tienen gran importancia por el papel que desempeña la Internet en el mundo actual. La plataforma WWW (World Wide Web) ha evolucionado paulatinamente para convertirse en un ambiente donde se implementan potentes aplicaciones cliente/servidor, unido a ello han ido surgiendo nuevas tecnologías que se relacionan con el desarrollo web lo que hacen a éste más interactivo e interesante (Madruga & Morera, 2006).

Sistemas cliente-servidor

El término cliente-servidor se refiere principalmente a una arquitectura o división lógica de responsabilidades. El cliente es la aplicación (conocida también como interfaz o parte frontal) y el servidor es el DBMS (conocida también como servidor de fondo o parte dorsal). Como claramente el sistema puede ser dividido en dos partes, es posible ejecutar a las dos partes en máquinas diferentes. Debido a que esta posibilidad es tan atractiva, el término cliente-servidor ha venido aplicándose casi exclusivamente para el caso donde el cliente y el servidor están en máquinas diferentes (Date, 2001).

Existen varias posibilidades de topología para esta arquitectura (Connolly & Begg, 2005; Date, 2001):

Varios clientes y un servidor

Un cliente puede acceder a varios servidores: o El cliente accede a un servidor a la vez o El cliente accede a varios servidores simultáneamente; es decir, una solicitud puede involucrar datos de varios servidores y el cliente lo ve como si fuera una sola unidad (sistema de base de datos distribuida).

Arquitectura cliente-servidor con dos capas

Según (Connolly & Begg, 2005) las aplicaciones con requerimientos de datos están formadas básicamente por cuatro componentes: la base de datos, la lógica de transacciones, la lógica del negocio, y la gestión de datos y la interfaz de usuario. La arquitectura cliente-servidor con dos capas (conocidas también como niveles, o los términos del inglés level o tier), proporciona una separación muy básica para estos componentes. El cliente es el responsable de la presentación de los datos al usuario, mientras que el servidor es quien los provee al cliente (ver Anexo 1.1).

En una interacción típica entre el cliente y el servidor, el cliente acepta la solicitud del usuario, comprueba la sintaxis y genera solicitudes de base de datos en SQL o en otro lenguaje que resulta apropiado para la lógica de la aplicación. Luego transmite el mensaje al servidor, espera la respuesta y formatea los resultados para el usuario final. El servidor acepta y procesa las solicitudes de base de datos y finalmente transmite los resultados al cliente.

A los clientes de la arquitectura con dos capas, también se les conoce como clientes pesados (del inglés fatclient) (Reese, 2000) o complejos (Connolly & Begg, 2005), pues contienen la interfaz de usuario y la lógica empresarial (reglas del negocio) de la aplicación (Connolly & Begg, 2005; Silberschatz, Korth, & Sudarshan, 2002). Los clientes pesados, por lo general, se conectan a través de una API con el DBMS (ver Anexo 1.2) que se encuentra en la máquina servidor (Elmasri & Navathe, 2004). Las API más ampliamente difundidas son ODBC para sistemas Windows y JDBC de la plataforma Java (Fisher, Ellis, & Bruce, 2003).

Para (Connolly & Begg, 2005) las dos capas presentan las siguientes ventajas:

Permite un acceso más universal a las bases de datos existentes.

Mejores prestaciones: si los clientes y el servidor residen en computadoras distintas, las diferentes CPU podrán procesar distintas aplicaciones en paralelo. También resulta más fácil optimizar la máquina servidora si su única tarea consiste en realizar el procesamiento relacionado con la base de datos.

Pueden reducirse los costos de hardware: sólo el servidor requiere el suficiente espacio de almacenamiento y la suficiente capacidad de proceso como para almacenar y gestionar la base de datos.

Se reducen los costes de comunicaciones: las aplicaciones llevan a cabo parte de las operaciones en el cliente y sólo envían a través de la red las solicitudes de acceso a la base de datos, con lo que se envían menos datos a través de la red.

Mayor coherencia: el servidor puede gestionar las comprobaciones de identidad, por lo que sólo es necesario definir y validar las restricciones en un único lugar, en lugar de hacer que cada programa de aplicación lleve a cabo sus propias comprobaciones.

1.4.1 Arquitectura cliente-servidor con tres capas

Como consecuencia de algunas limitaciones en la arquitectura con dos capas, se propuso añadir una nueva capa intermedia (conocida también por los términos del

inglés middle-tier o middleware). La solución se concentró fundamentalmente en el aumento de la escalabilidad y del mantenimiento del código con menor impacto.

A mediados de la década de 1990, a medida que las aplicaciones crecieron en complejidad y que el número de clientes para un servidor aumentó, la escalabilidad disminuyó apreciablemente, lo cual se mostró a través de dos síntomas (Connolly & Begg, 2005):

Clientes complejos que necesitaban mayores recursos (RAM, espacio de almacenamiento y potencia de procesamiento) en las máquinas clientes.

En adición, el mantenimiento del código de las aplicaciones en dos capas era menos flexible debido a que la conexión con el DBMS no estaba aislado del cliente. Si se cambiaba el esquema de base de datos, entonces era necesario cambiar la conexión con la base de datos en la aplicación cliente, lo cual obligaba a su recompilación y redistribución. Por otra parte, como la lógica del negocio también estaba en la parte cliente, cada nueva aplicación cliente (vista de usuario) debía volver a implementarla (Reese, 2000).

La arquitectura en tres capas soluciona los problemas de escalabilidad y mantenimiento del código al añadir la capa intermedia. En esta nueva capa, inserta la lógica del negocio y la conexión con el DBMS. Los clientes se tornan ligeros o sencillos, pues solo se encargan de la presentación de los datos al usuario (interfaz). Ahora, un cambio en el esquema de bases de datos solo afecta la capa intermedia, la cual contiene la conexión (ver Anexo 1.3). Así, se evita el gran problema de la recopilación y redistribución de las aplicaciones cliente. Por su parte, las diferentes aplicaciones cliente (vistas de usuario) se sirven de la misma lógica empresarial, por lo que el procesamiento de las reglas del negocio es ahora centralizado en la capa intermedia (Connolly&Begg, 2005; Post, 2005; Reese, 2000).

Por tanto, según se apunta en (Silberschatz et al., 2002), la principal distinción con los sistemas en dos capas radica en si la aplicación cliente accede directamente al DBMS (caso de los de dos capas) o si delega en una capa intermedia (caso de los de tres capas). Así, un sistema cliente-servidor con tres capas es se compone de tres partes, niveles o capas.

Nótese que los sistemas cliente-servidor responden a una división lógica de responsabilidades, por lo que pueden representar componentes de software que residen en una misma máquina o en varias. De la misma manera, los

componentes del sistema cliente-servidor con tres capas pueden distribuirse de varias formas. Por ejemplo, es posible que tanto el servidor de aplicaciones como el de base de datos estén en la misma máquina. No obstante, muchas empresas ubican al servidor de aplicaciones separado del servidor de base de datos para aumentar el rendimiento mediante el balance de cargas entre ambas máquinas (Connolly & Begg, 2005; Núñez Camallea & Coutin Abalo, 2005).

En los principios de la Internet, las páginas web tenían muy poca gestión de datos y se componían de páginas estáticas resueltas por el servidor web y visualizadas por el explorador web. En tales casos, la arquitectura cliente-servidor en dos capas era suficiente. En la actualidad, la mayoría de los sitios que albergan aplicaciones web implementan, por naturaleza y no por decisión del programador, la arquitectura cliente-servidor en tres capas. En (Connolly&Begg, 2005, p. 55) se dice que “la arquitectura en tres niveles se adapta de forma bastante natural a los entornos web”, lo cual es corroborado por otros autores (Silberschatz et al., 2002, p. 12). Más adelante agrega que tal arquitectura es “adecuada para algunos entornos como la Internet y las intranets corporativas, en las que puede utilizarse un explorador web como cliente”(Connolly&Begg, 2005, p. 56).

1.5 Patrón MVC

Dentro de la arquitectura de n capas se encuentra el patrón Modelo Vista Controlador (MVC). El MVC es una aproximación al software que separa la lógica de la aplicación de la presentación.

En la práctica, permite que sus páginas web contengan mínima codificación ya que la presentación es separada del código (Viktor, 2010).

El modelo representa la estructura de datos. Típicamente sus clases de modelo contendrán funciones que lo ayudarán a recuperar, insertar y actualizar información en su base de datos.

La vista es la información que es presentada al usuario. La vista normalmente será una página web, también puede ser un fragmento de una página como un encabezado.

El controlador sirve como un intermediario entre el modelo, la vista y cualquier otro recurso necesario para procesar la petición y generar una página web.

Presenta ventajas como:

Se consiguen múltiples vistas simultáneamente, pues la vista está separada del modelo.

La conexión entre el modelo y sus vistas es dinámica; se produce en tiempo de ejecución, no en tiempo de compilación.

No acoplamiento, lo que posibilita la facilidad de evolución. Las interfaces de usuario pueden cambiar sin afectar las reglas del negocio o los datos (Viktor, 2010).

Por lo anteriormente expuesto se considera que el MVC ofrece grandes ventajas en la creación de un sitio web como el que se desea implementar en la presente investigación.

1.6 Tecnologías del lado del cliente

Las tecnologías del lado del cliente están insertadas en la página HTML (HyperText Markup Language) del cliente y son interpretadas y ejecutadas por el navegador. Es decir, que su correcta funcionalidad depende del soporte de la versión del navegador a ser utilizado por el usuario visitante (M. Rodríguez & Alejo, 2006).

HTML

El HTML es un formato de datos para crear documentos de hipertexto que puedan ser entendidos desde computadoras con diferentes sistemas operativos. Consiste en un sencillo sistema de marcas, que insertadas en un texto, permiten el enlace con otras páginas o recursos (imágenes, sonidos, entre otros.) y establecen ciertos atributos del documento (dimensiones, alineamiento del texto, estilo, y otros.). La flexibilidad que posee es tal que sus características pueden aprovecharse para generar documentos multimedia, aunque no vayan a colocarse en un servidor (Vidal, 2006).

HTML es un lenguaje de marcas que sigue una definición formal de acuerdo con un estándar.

Su potencia se basa en su independencia del sistema operativo y que identifica cada parte de un documento por la finalidad que tiene (Vidal, 2006).

No tiene la complejidad de un lenguaje de programación, pues es bastante sencillo de utilizar y no precisa de ningún compilador. Más bien, es semejante a un documento de texto. Todo documento HTML es almacenado como texto normal en

formato ASCII, de manera que cualquier procesador o editor de textos puede ser utilizado para su creación. El navegador es quien se encarga de leer ese documento de texto e interpretar las etiquetas HTML que aparecen en él (Vidal, 2006).

Dentro de sus principales ventajas (Cadenas, 2010):

Sencillo, que permite describir hipertexto.

Texto presentado de forma estructurada y agradable.

No se necesita de grandes conocimientos cuando se cuenta con un editor de páginas web o WYSIWYG (What You See Is What You Get).

Despliegue rápido.

Lenguaje de fácil aprendizaje.

Lo admiten todos los exploradores.

Y sus desventajas (Cadenas, 2010):

Lenguaje estático.

La interpretación de cada navegador puede ser diferente.

Guarda muchas etiquetas que pueden convertirse en “basura” y dificultan la corrección.

El diseño es más lento.

Las etiquetas son muy limitadas.

JavaScript

Es un lenguaje de tipo script compacto, basado en objetos y guiado por eventos, diseñado específicamente para el desarrollo de aplicaciones cliente/servidor dentro del ámbito de Internet. Ofrece la posibilidad de agregar un poco de dinamismo y de interactividad en un sitio web (Vedora Willock, 2006).

Este lenguaje proporciona los medios para controlar las ventanas del navegador y el contenido que muestran. Permitiendo crear contenidos dinámicos y efectos especiales en las páginas tales como: dar movimiento a determinados elementos de la página, cambiarles el color, abrir ventanas, generar ventanas de diálogo, realizar cambio de imágenes, menús desplegados, animar capas y otros dinamismos. Y evitar depender del servidor web para cálculos sencillos, validación de datos que el usuario introduce en un formulario antes de enviarlos, responder a eventos generados por el usuario (Vedora Willock, 2006).

Sus principales características según Vedora Willock (2006) es un lenguaje interpretado puro (no requiere compilación, ni generación de intermedios

codificados de ningún tipo). El navegador del cliente es el encargado de interpretar las instrucciones JavaScript y ejecutarlas.

Está basado en Objetos y guiado por Eventos. También es sencillo y pensado para hacer las tareas con rapidez. Incluso personas que no tengan una experiencia previa en la programación podrán aprender este lenguaje con facilidad y utilizarlo en toda su potencia con solo un poco de práctica. Y puede ejecutarse en el cliente evitando la necesidad de realizar peticiones continuamente al servidor, por lo que descarga el trabajo de este y hace que la navegación sea más rápida.

Sus ventajas, expresa Cadenas (2010) son:

Lenguaje de scripting seguro y fiable.

Los script tienen capacidades limitadas, por razones de seguridad.

El código Java Script se ejecuta en el cliente.

Y por el mismo autor sus desventajas son:

Código visible por cualquier usuario.

El código debe descargarse completamente.

CSS

CSS (Cascade Style Sheets) es una tecnología que permite controlar la presentación de los documentos en la Web. Las Hojas de Estilo en Cascada o CSS constituyen un lenguaje sencillo que complementa el de HTML, suponiendo un apoyo fundamental a la hora de diseñar páginas web, porque permiten una mayor precisión en el ajuste de los elementos de diseño (Vedora Willock, 2006).

Esta técnica consiste en separar el diseño del contenido, de manera que las indicaciones para conformar el diseño se agrupan en una hoja de estilo o archivo fuera del contenido del documento de la página HTML. Lo que hace fundamentalmente el código de las hojas de estilos es transformar las etiquetas del lenguaje HTML y conformarlas a las características que se quiera darle; pero también, con este código se pueden crear etiquetas nuevas, que se introducen dentro del documento. Una de las ventajas de las hojas de estilos es que se puede modificar algunas características de todos los documentos de un sitio web desde un archivo, sin tener que modificarlas en cada uno de los documentos (Alonso & Hernández, 2006).

El CSS es la mejor forma de separar los contenidos y su presentación, es imprescindible para crear páginas web complejas. Reduce la complejidad de su mantenimiento. Como también permite visualizar el mismo documento en infinidad

de dispositivos diferentes. Y especifica la apariencia de todo el sitio web en un único lugar, los estilos para fuentes, tamaños de la fuente, espaciado de letras, colores, y márgenes. Un cambio en un fichero genera cambios en todos los ficheros. Según las ventajas de CSS encontradas en Cadenas (2010). Y como decadencias plantea que no todos los navegadores soportan CSS.

1.7 Tecnologías del lado del servidor

Las tecnologías del lado del servidor pueden o no estar insertadas dentro de la página HTML. A diferencia de las tecnologías del lado del cliente, estas no dependen del navegador ya que son interpretadas y ejecutadas por el servidor. Por ejemplo, si se utiliza PHP en un sitio determinado se necesita que el servidor donde esté alojado el mismo, tenga instalado PHP (M. Rodríguez & Alejo, 2006).

PHP

PHP (siglas que originalmente significaban Personal Home Page, pero actualmente significan Hypertext Preprocessor) es un lenguaje de programación el cual se ejecuta en los servidores web y que permite crear contenido dinámico en las páginas HTML. Dispone de múltiples herramientas que permiten acceder a bases de datos de forma sencilla, por lo que es ideal para crear aplicaciones para Internet. Es multiplataforma, de forma que el código que se haya creado para una de ellas no tiene por qué modificarse al pasar a la otra. Es un lenguaje de programación de estilo clásico, con variables, sentencias condicionales, bucles, funciones, entre otras. La sintaxis que utiliza, la toma de otros lenguajes muy extendidos como C y Perl. El código de PHP está incluido en tags especiales "<?php, ?>" («Introducción a PHP», s. f.).

En ningún caso se envía código PHP al navegador, por lo que todas las operaciones realizadas son transparentes al usuario, el código PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML. Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP («Introducción a PHP», s. f.).

El PHP tiene varias ventajas dentro de estas Cadenas (2010) expone que es muy fácil de aprender, se caracteriza por ser un lenguaje muy rápido, soporta en cierta

medida la orientación a objeto, clases y herencia. También que tiene capacidad de conexión con la mayoría de los manejadores de base de datos: MySQL, PostgreSQL, Oracle, MS SQL Server, entre otras. Es libre, por lo que se presenta como una alternativa de fácil acceso para todos. Incluye gran cantidad de funciones. No requiere definición de tipos de variables ni manejo detallado del bajo nivel. Pero tiene en contra que depende de la instalación de un servidor web y todo el trabajo lo realiza el servidor sin delegar al cliente.

1.8 Servidores web

Un servidor web es un programa que implementa el protocolo HTTP (Hypertext Transfer Protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas

web o páginas HTML, textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos. Entre los servidores más utilizados en Internet se encuentran Apache e IIS (Internet Information Services) (Vedora Willock, 2006).

Servidor web Apache

Apache tal vez no sea uno de los servidores web más fáciles de configurar, pero una vez instalado, Apache se convertirá en el servidor más potente y estable que se pueda imaginar. Es Open Source, o sea código abierto, lo que significa que su código es libre y gratuito. Además, es el servidor más utilizado en Internet (Ruiz, 2006).

Este servidor está instalado por defecto en todas las distribuciones Linux. También existe para otras plataformas, incluso Windows. Su funcionamiento básico es ejecutando un proceso padre y tantos procesos hijos como peticiones reciba para atender a cada cliente. Apache es usado para enviar páginas web estáticas y dinámicas en la WWW. Muchas aplicaciones web están diseñadas asumiendo como ambiente de implantación a Apache, o utilizarán características propias de este servidor web.

Ventajas (Cadenas, 2010).

Multiplataforma

El software es gratuito.

Estándar del sector para la mayoría de servidores web.

Permite la administración remota.

Tiene interfaz con todos los sistemas de autenticación.

Tiene integración estándar del protocolo de seguridad SSL.

Provee interfaz a todas las bases de datos.

Desventajas (Cadenas, 2010).

Instalación en modo consola.

Apache no se actualiza regularmente.

Requiere un conocimiento más técnico para instalar y configurar.

1.9 Sistema Gestores de Base de Datos

Una Base de Datos (BD) es un conjunto de datos interrelacionados, almacenados con carácter más o menos permanente en la computadora, puede ser considerada una colección de datos variables en el tiempo (Matos, 2004, p. 10).

Un Sistema de Gestión de Base de Datos (SGBD) es el software que permite la utilización y la actualización de los datos almacenados en una o varias bases de datos desde diferentes puntos de vista y a la vez. Su objetivo fundamental es suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, de forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora, ni el método de acceso empleado. Cumple los objetivos de garantizar independencia de los datos y los programas de aplicación, minimizar las redundancias, integrar y sincronizar las bases de datos y los datos que estas contienen, al tiempo que los protege y facilita la manipulación de la información, mediante un control centralizado. La información es representada a través de tuplas, las cuales describen al fenómeno, proceso o ente de la realidad objetiva que se está analizando y se representan a través de tablas (Alonso & Hernández, 2006).

En la actualidad existen muchos de SGBD, muchos de ellos muy potentes, entre los que se encuentran Microsoft SQL Server, Oracle, SQL, MySQL, entre otros. En esta investigación se hace especial alusión al PostgreSQL, por las características que se exponen a continuación y que demuestran que puede resultar una buena elección a la hora de concebir la capa de datos de cualquier arquitectura que implemente el modelo de N capas.

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional (ORDBMS) basado en el proyecto POSTGRES, de la universidad de Berkeley. El director de este proyecto es el profesor Michael Stonebraker, y fue patrocinado por

Defense Advanced Research Projects Agency (DARPA), el Army Research Office (ARO), el National Science Foundation (NSF), y ESL, Inc.

PostgreSQL es una derivación libre (OpenSource) de este proyecto, y utiliza el lenguaje SQL92/SQL99, así como otras características que comentaremos más adelante.

Fue el pionero en muchos de los conceptos existentes en el sistema objeto-relacional actual, incluido, más tarde en otros sistemas de gestión comerciales. PostgreSQL es un sistema objeto-relacional, ya que incluye características de la orientación a objetos, como puede ser la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. A pesar de esto, PostgreSQL no es un sistema de gestión de bases de datos puramente orientado a objetos.

Historia de PostgreSQL

PostgreSQL (llamado también Postgres95) fue derivado del proyecto Postgres, como ya se ha comentado. A sus espaldas, este proyecto lleva más de una década de desarrollo, siendo hoy en día, el sistema libre más avanzado con diferencia, soportando la gran mayoría de las transacciones SQL, control concurrente, teniendo a su disposición varios “language bindings” como por ejemplo C, C++, Java, Python, PHP y muchos más.

La implementación de Postgres DBMS comenzó en 1986, y no hubo una versión operativa hasta 1987. La versión 1.0 fue liberada en Junio de 1989 a unos pocos usuarios, tras la cual se liberó la versión 2.0 en Junio de 1990 debido a unas críticas sobre el sistema de reglas, que obligó a su reimplementación. La versión 3.0 apareció en el año 1991, e incluyó una serie de mejoras como una mayor eficiencia en el ejecutor de peticiones. El resto de versiones liberadas a partir de entonces, se centraron en la portabilidad del sistema. El proyecto se dio por finalizado en con la versión 4.2, debido al gran auge que estaba teniendo, lo cual causó la imposibilidad de mantenimiento por parte de los desarrolladores.

En 1994, Andrew Yu y Jolly Chen añadieron un intérprete de SQL a este gestor. Postgres95, como así se llamó fue liberado a Internet como un proyecto libre (OpenSource). Estaba escrito totalmente en C, y la primera versión fue un 25%

más pequeña que Postgres, y entre un 30 y un 50% más rápida. A parte de la corrección de algunos bugs, se mejoró el motor interno, se añadió un nuevo programa monitor, y se compiló usando la utilidad GNU Make y el compilador gcc sin necesidad de parchearlo (como había hecho falta en versiones anteriores).

En 1996, los desarrolladores decidieron cambiar el nombre a al DBMS, y lo llamaron PostGreSQL (versión 6.0) para reflejar la relación entre Postgres y las versiones recientes de SQL. Se crearon nuevas mejoras y modificaciones, que repercutieron en un 20-40% más de eficiencia, así como la incorporación del estándar SQL92.

A continuación se enumeran las principales características de este gestor de bases de datos:

1. Implementación del estándar SQL92/SQL99.
2. Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP ...), cadenas de bits, etc. También permite la creación de tipos propios.
3. Incorpora una estructura de datos array.
4. Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.
5. Permite la declaración de funciones propias, así como la definición de disparadores.
6. Soporta el uso de índices, reglas y vistas.
7. Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
8. Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

PostGreSQL es un magnífico gestor de bases de datos, capaz de competir con muchos gestores comerciales, aunque carezca de alguna característica casi imprescindible. Ésta es, bajo mi punto de vista, un conjunto de herramientas que permitan una fácil gestión de los usuarios y de las bases de datos que contenga el sistema. Por otro lado, la velocidad de respuesta que ofrece este gestor con bases

de datos relativamente pequeñas puede parecer un poco deficiente, aunque esta misma velocidad la mantiene al gestionar bases de datos realmente grandes, cosa que resulta loable.

El lenguaje PHP posee un amplio conjunto de funciones definidas para el tratamiento de este gestor.

La selección de estas tecnologías se basó principalmente en los criterios de rapidez, sencillez y rendimiento, teniendo en cuenta su tecnología Open Source y la característica de integrarse para funcionar en cualquier plataforma. Estas características aportan un gran beneficio gracias al entorno de software libre de la actualidad y cumplen con una política trazada por el país de proyectarse con visión al futuro y comenzar a desarrollar este tipo de aplicaciones. Como SGBD se seleccionó PostgreSQL, por su velocidad y sencillez. Se seleccionó la utilización de PHP, ya que esta tecnología exhibe un magnífico soporte para el trabajo con PostgreSQL.

Finalmente como servidor web se determinó que la mejor opción era el Apache, por mostrar un excelente comportamiento y estabilidad. Este servidor es muy utilizado en el mundo por su magnífico desempeño, seguridad y las posibilidades de gestionar de una forma eficiente muchos de los aspectos que definen en gran medida la calidad de los servidores web. La selección de Apache añade una integración perfecta con las tecnologías antes mencionadas.

Apache-PHP-PostgreSQL forman en su conjunto una vía muy eficaz a la hora de tener en cuenta la implementación de sitios web de corto, mediano o gran alcance.

1.10 Lenguaje de Modelado Unificado (UML)

El UML (Unified Modeling Language) permite modelar, construir y documentar los elementos que forman un producto de software que responde a un enfoque orientado a objetos. Este lenguaje fue creado por un grupo de estudiosos de la Ingeniería de Software formado por: Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1995. Desde entonces, se ha convertido en el estándar internacional para definir, organizar y visualizar los elementos que configuran la arquitectura de una aplicación orientada a objetos. Con este lenguaje, se pretende unificar las experiencias acumuladas sobre técnicas de modelado e incorporar las mejores prácticas actuales en un acercamiento estándar. No es un lenguaje de

programación sino un lenguaje de propósito general para el modelado orientado a objetos y también puede considerarse como un lenguaje de modelado visual que permite una abstracción del sistema y sus componentes (Sosa & Ortiz, 2006).

Pressman (2007) señala que entre sus objetivos fundamentales se encuentran:

Ser tan simple como sea posible, pero manteniendo la capacidad de modelar toda la gama de sistemas que se necesita construir.

Necesita ser lo suficientemente expresivo para manejar todos los conceptos que se originan en un sistema moderno, tales como la concurrencia y distribución, así como también los mecanismos de la ingeniería de software, como son el encapsulamiento y el uso de componentes.

Debe ser un lenguaje universal, como cualquier lenguaje de propósito general.

Imponer un estándar mundial. La decisión de utilizar UML como notación para el desarrollo del sitio web se debe a que se ha convertido en un estándar que tiene las siguientes características:

Permite modelar sistemas utilizando técnicas orientadas a objetos.

Permite especificar todas las decisiones de análisis y diseño, construyéndose así modelos precisos, no ambiguos y completos.

Puede conectarse con lenguajes de programación (Ingeniería directa e inversa).

Permite documentar todos los artefactos de un proceso de desarrollo (requisitos, arquitectura, pruebas, versiones, entre otros.).

Es un lenguaje muy expresivo que cubre todas las vistas necesarias para desarrollar y luego desplegar los sistemas.

Existe un equilibrio entre expresividad y simplicidad, pues no es difícil de aprender ni de utilizar.

Es independiente del proceso, aunque para utilizarlo óptimamente se debería usar en un proceso que fuese dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental.

1.10.1 Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado de Desarrollo (RUP, por su denominación en inglés Rational Unified Process), fue creado por el mismo grupo de expertos que crearon UML, Ivar Jacobson, Grady Booch y James Rumbaugh en el año 1998. El objetivo que se perseguía era producir software de alta calidad, es decir, que cumpla con los requerimientos de los usuarios dentro de una planificación y presupuesto establecidos. Esta metodología concibió desde sus inicios el uso de UML como

lenguaje de modelado. Es un proceso dirigido por casos de uso, este avanza a través de una serie de flujos de trabajo, está centrado en la arquitectura y es iterativo e incremental. Además cubre el ciclo de vida de desarrollo de un proyecto y toma en cuenta las mejores prácticas a utilizar en el modelo de desarrollo de software (Pressman, 2007).

A continuación se muestran estas prácticas (Díaz & Angélica, 2004).

Desarrollo de software en forma iterativa.

Manejo de requerimientos.

Utiliza arquitectura basada en componentes.

Modelación del software visualmente

Verifica la calidad del software.

Controla los cambios.

Para apoyar el trabajo con esta metodología ha sido desarrollada por la compañía norteamericana Rational Corporation la herramienta CASE (Computer Aided Assisted Automated Software Engineering) Rational Rose. Esta herramienta integra todos los elementos que propone la metodología para cubrir el ciclo de vida de un proyecto.

1.11 Herramientas de desarrollo

1.11.1 El editor de texto CKEditor

El CKEditor es un editor de texto HTML / WYSIWYG de código abierto que provee a la Web del poder de las aplicaciones de escritorio al estilo de editores como Microsoft Word, sin la necesidad de instalar ningún componente en la computadora del cliente. Permite además la generación de código XHTML 1.0, soporte CSS, incorporación de formularios, formateo de fuente, cortar, copiar, pegar, inserción de imágenes, creación de tablas, menús contextuales con botón derecho, entre otros. Soporta sistemas operativos tales como Windows, MacOS y Linux con los lenguajes de servidor PHP, Perl, Python, ASP.NET ASP, ColdFusion, Java, Active-FoxPro , Lasso entre otros (Knabben, 2009).

1.11.2 Macromedia Dreamweaver

Dreamweaver es una herramienta de diseño de páginas web muy avanzada, tal como se afirma en muchos medios. Aunque sea un experto programador de HTML el usuario que lo maneje, siempre se encontrará en este programa razones para utilizarlo, sobre todo en lo que a productividad se refiere (Alvarez et al., 2005).

Cumple perfectamente el objetivo de diseñar páginas con aspecto profesional, y soporta gran cantidad de tecnologías, además muy fácil de usar: hojas de estilo y capas, Java Script para crear efectos e interactividades e inserción de archivos multimedia.

Visual Paradigm Visual Paradigm es una herramienta CASE: Ingeniería de Software Asistida por Computación. La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación.

Visual Paradigm ha sido concebida para soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de todo tipo de diagramas. Constituye una herramienta privada disponible en varias ediciones, cada una destinada a satisfacer diferentes necesidades: Enterprise, Professional, Community, Standard, Modeler y Personal. Existe una alternativa libre y gratuita de este software, la versión Visual Paradigm UML 6.4 Community Edition (Community Edition, ya que existe la Enterprise, Professional, etc). Fue diseñado para una amplia gama de usuarios interesados en la construcción de sistemas de software de forma fiable a través de la utilización de un enfoque Orientado a Objetos. Las principales características porque en nuestra investigación utilizamos el Visual Paradigm son por su disponibilidad en múltiples plataformas (Windows, Linux). Su diseño está centrado en casos de uso y enfocado al negocio que genera un software de mayor calidad. Utiliza un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación. Presenta capacidades de ingeniería directa e inversa. Su licencia es gratuita y comercial. Soporta aplicaciones Web. Las imágenes y reportes generados, no son de muy buena calidad. Es un software fácil de instalar y actualizar. Cuenta con los diagramas de Procesos de Negocio - Proceso, Decisión, Actor de negocio, Documento. Puede hacer ingeniería inversa de bases de datos Desde Sistemas Gestores de Bases de Datos (DBMS) existentes a diagramas de Entidad-Relación. Y por último pude hacer una distribución automática de diagramas.

Conclusiones parciales del capítulo

De acuerdo al estudio teórico y metodológico realizado se determinó la necesidad de crear una aplicación web para contribuir a la eficiencia del proceso de enseñanza aprendizaje de la ciencia Química como asignatura del octavo grado de la enseñanza preuniversitaria en Cuba. De esta manera, el proceso de automatización de la información permitirá una mejor manipulación y divulgación de los materiales referentes a la misma, así como el fácil acceso por parte de los usuarios que interactúen con ella, con vistas a que el proceso de enseñanza – aprendizaje sea más ameno, eficaz y motivador para estudiantes y profesores.

Se adoptó al RUP como metodología de desarrollo de sistemas, con el UML para el modelado asistido por diagramas.

Combinado con el HTML para estructurar las páginas web, se utilizó el lenguaje Java Script, el cual provee el dinamismo necesario. La presentación de las diferentes páginas, se elaboró con CSS; y para implementar la lógica del problema, se utilizó PHP.

Entre las arquitecturas físicas, se tomó a la de dos capas como la más adecuada debido a la experiencia en su manejo y a sus consabidos beneficios. De manera lógica, el código de la implementación se organizó en tres partes interrelacionadas mediante el patrón de diseño conocido como MVC.

Por las ventajas evidenciadas en el estudio se determinó utilizar como servidor web el “Wamp”, gestor de base de datos “PostgreSQL”, y para el modelado de la ingeniería “Visual Paradigm”.

Capítulo II: Diseño de una Aplicación Web que contribuya a mejorar el aprendizaje de la asignatura Química en onceno grado.

En este capítulo se exponen las reglas del negocio a tener en cuenta para el diseño del sistema, una caracterización de los actores y trabajadores del mismo, sus casos de uso, conformándose el diagrama de casos de uso, la descripción de estos en formato expandido, y se muestran los diagramas de actividades y del modelo objetos.

Además se describe de modo general el funcionamiento del sistema diseñado. Se definen los requerimientos funcionales y no funcionales, se realiza la descripción del modelo de casos de uso del sistema, basándose en los actores y los casos de uso.

2.1 Modelo del negocio

Cuando se hable de procesos de negocio se puede decir que son un grupo de tareas relacionadas lógicamente que se llevan a cabo en una determinada secuencia y forma, que emplean los recursos de la organización para dar resultados que apoyen sus objetivos. (Jacobson, Booch, & Rumbaugh, 2006).

A partir de la definición del proceso del negocio se realizó un estudio sobre la enseñanza preuniversitaria en nuestra provincia y se detectó que existen deficiencias en relación a la informatización de asignaturas en cuestión, por lo que después de identificado el proceso para ser modelado se escogió el proceso de enseñanza aprendizaje de la Química como asignatura del onceno grado en la enseñanza media superior.

2.1.2 Reglas del negocio a considerar

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio (Jacobson, Booch, & Rumbaugh, 2006).

Partiendo de lo planteado anteriormente fueron identificadas las siguientes reglas:

1. Para tener acceso a la información y herramientas que ofrece el sistema es necesario que la persona se encuentre registrada como usuario del mismo.

2. La única persona que tiene autorización para eliminar contenido o cuentas de usuario del sistema así como de crear un usuario con privilegios administrativos es el profesor de la asignatura y el jefe de departamento.
3. La inserción de datos en el sistema va a ser realizada por el profesor, el jefe de departamento y colectivo de la asignatura, estos datos deben cumplir con la estructura para el tipo de información que se desea insertar.
4. El profesor debe orientarle ejercicios y ejemplos al estudiante para el desarrollo de sus conocimientos.
5. El estudiante debe consultar la bibliografía para la realización realizar el estudio.
6. El profesor debe evaluar los conocimientos del estudiante.

2.1.3 Modelado del negocio

El modelo de Casos de Uso del Negocio es el encargado de describir los procesos de una empresa utilizando los casos de uso y los actores, en correspondencia, a su vez, con los procesos del negocio y los clientes (Jacobson, Booch, & Rumbaugh, 2006).

El modelo de Casos de Uso del Negocio es definido a través de tres artefactos:

1. El diagrama de casos de uso del negocio,
2. La descripción de los casos de uso del negocio
3. El diagrama de actividades de cada caso de uso del negocio.

2.1.4 Actores del Negocio

Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados. (Jacobson, Booch, & Rumbaugh, 2006)

Actor	Descripción
Estudiante	Para el estudio, el estudiante debe consultar la bibliografía, que consiste en libros de texto y algunos materiales complementarios dispersos.

Tabla 1: Actor del negocio

2.1.5 Trabajadores del Negocio

Un trabajador es una abstracción de una persona (o grupo de personas), una máquina o un sistema automatizado; que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores y manipulando entidades. (Jacobson, Booch, & Rumbaugh, 2006)

Trabajador del Negocio	Justificación
Profesor	Es quien se encarga de la gestión de la información que solicita el estudiante.

Tabla 2 Trabajador del negocio

2.1.6 Diagrama de Casos de Uso del Negocio

Un diagrama de casos de uso representa gráficamente a los procesos del negocio como casos de usos y su interacción con los actores. (Jacobson, Booch, & Rumbaugh, 2006)

Para tener una visión general de los procesos del negocio en la entidad se confecciona el diagrama de Casos de Uso del Negocio.

2.1.7 Descripción de los Casos de Uso del negocio

Luego de identificar el proceso del negocio es necesario explicarlo detalladamente. A continuación se muestra la descripción del mismo.

Tabla 3 Descripción del Caso de Uso del Negocio.

Caso de Uso del Negocio	Consultar bibliografía
Actor	Estudiante
Propósito	Consultar la bibliografía orientada para realizar el estudio.
Resumen El caso de uso se inicia cuando el profesor le orienta al alumno el estudio independiente y este revisa la bibliografía que se encuentra a su disposición en ese momento (libro de texto, notas de clases y algunos materiales de apoyo dispersos).	
Curso Normal de los eventos	

Acción del Actor		Respuesta del negocio
		1. El Profesor orienta la bibliografía a consultar.
2. El Estudiante revisa el libro de texto, notas de clases y algunos materiales de apoyo dispersos.		
Prioridad	alta	

2.1.8 Diagrama de actividades

El diagrama de actividad es un grafo que contiene los estados en que puede hallarse la actividad a analizar. Cada estado de la actividad representa la ejecución de una sentencia de un procedimiento, o el funcionamiento de una actividad en un flujo de trabajo. En resumen describe un proceso que explora el orden de las actividades que logran los objetivos del negocio. (Jacobson, Booch, & Rumbaugh, 2006).

Para un mejor entendimiento del proceso en cuestión se realiza el Diagrama de Actividad correspondiente al Caso de Uso del Negocio Solicitar Información. El diagrama de actividad se presenta a continuación.

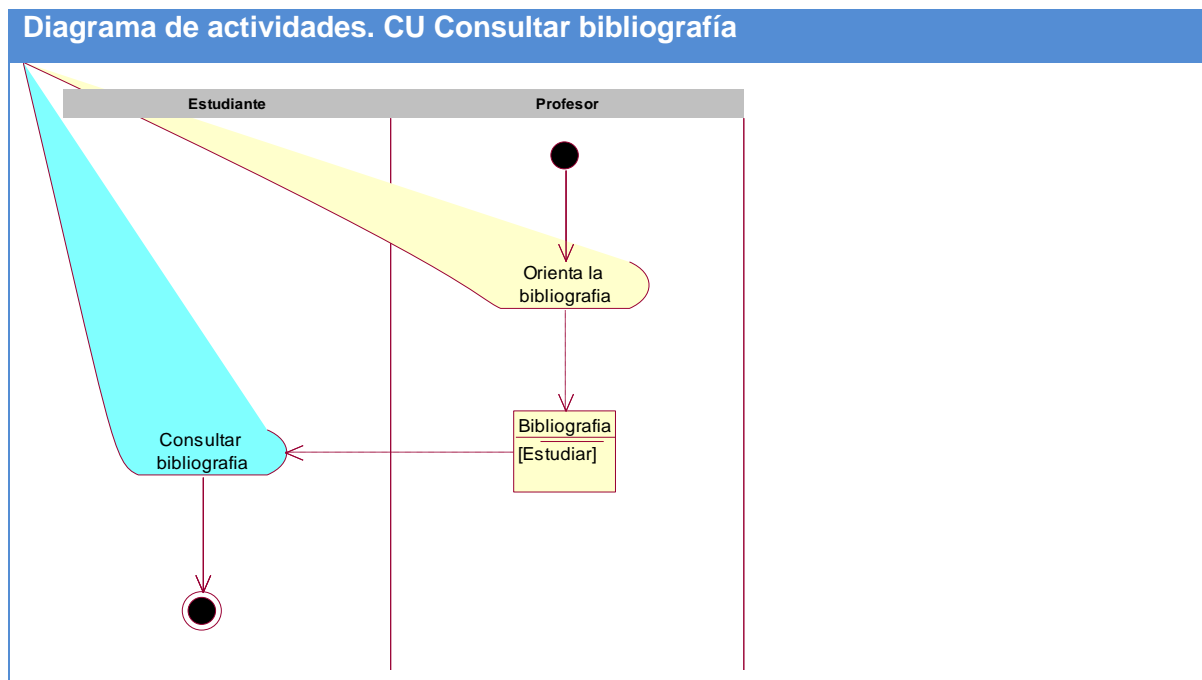


Figura 2. Diagrama de actividades. CU Consultar bibliografía.

2.1.9 Modelo de objetos del negocio

Un modelo de objetos del negocio es un modelo interno a un negocio. Describe como cada caso de uso del negocio es llevado a cabo por parte de un conjunto de trabajadores que utilizan un conjunto de entidades del negocio y unidades de trabajo. (Jacobson, Booch, & Rumbaugh, 2006)

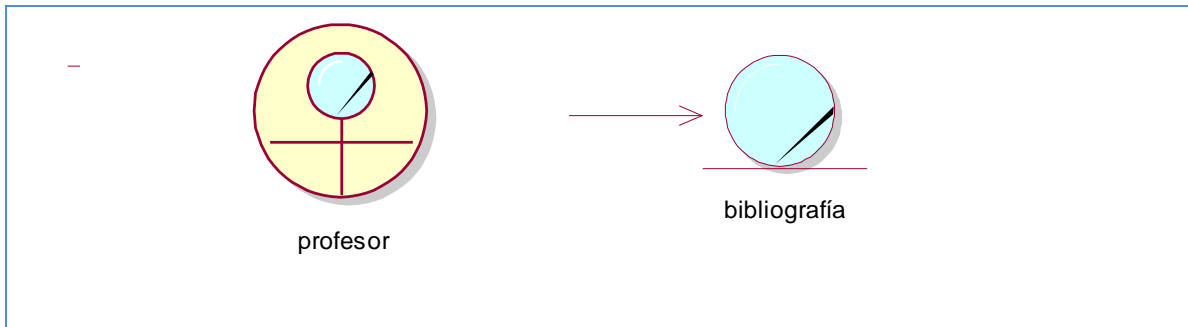


Figura 3. Diagrama del modelo de objetos.

2.2 Especificación de los requisitos de software

2.2.1 Requerimientos funcionales

Los requerimientos funcionales permiten expresar una especificación más detallada de las responsabilidades del sistema que se propone. Ellos permiten determinar, de una manera clara, lo que debe hacer el mismo. (Jacobson, Booch, & Rumbaugh, 2006)

Teniendo en cuenta lo planteado, el sistema propone los siguientes requerimientos funcionales:

Requerimientos Funcionales

1. Iniciar sesión

1.1 Entrar al sistema.

2. Autenticarse

2.1 Validar el nombre del usuario y su clave de acceso.

2.2 Validar que el usuario que se autentica tenga los permisos en el módulo de administración para introducir u obtener información del sistema.

3. Cambiar contraseña el usuario.

4. Gestionar usuario

4.1 4.1 Agregar nuevo usuario

4.2 Modificar usuario.

4.3 Eliminar usuario.

5. Gestionar estudiante

5.1 Agregar nuevo estudiante

5.2 Modificar estudiante.

5.3 Eliminar estudiante.

6. Gestionar asignatura.

6.1 Adicionar asignatura.

6.2 Modificar asignatura.

6.3 Eliminar asignatura.

7. Mostrar asignatura.

8. Gestionar programa.

8.1 Adicionar programa.

8.2 Modificar programa.

8.3 Eliminar programa.

9. Mostrar programa.

10. Gestionar unidad.

10.1 Adicionar unidad.

10.2 Modificar unidad.

10.3 Eliminar unidad.

11. Mostrar unidad.

12. Gestionar epígrafe.

12.1 Adicionar epígrafe.

12.2 Modificar epígrafe.

12.3 Eliminar epígrafe.

13. Mostrar epígrafe.

14. Gestionar bibliografía.

14.1 Adicionar bibliografía.

14.2 Modificar bibliografía.

14.3 Eliminar bibliografía.

15. Mostrar bibliografía.

16. Gestionar ejercicios.

16.1 Adicionar ejercicios.

16.2 Modificar ejercicios.

16.3 Eliminar ejercicios.

17. Mostrar ejercicios.

18. Gestionar material complementario.

18.1 Adicionar material complementario.

18.2 Modificar material complementario.

18.3 Eliminar material complementario.

19. Mostrar material complementario.

20. Gestionar calendario.

20.1 Adicionar calendario.

20.2 Modificar calendario.

20.3 Eliminar calendario.

21. Mostrar material calendario.

22. Gestionar enlaces.

22.1 Adicionar enlaces.

22.2 Modificar enlaces.

22.3 Eliminar enlaces.

23. Mostrar material enlaces.

24. Gestionar noticias.

24.1 Adicionar noticias.

24.2 Modificar noticias.

24.3 Eliminar noticias.

25. Mostrar material noticias.

26. Gestionar eventos.

26.1 Adicionar eventos.

26.2 Modificar eventos.

26.3 Eliminar eventos.

27. Mostrar material eventos.

28. Gestionar imágenes.

28.1 Adicionar imágenes.

28.2 Modificar imágenes.

28.3 Eliminar imágenes.

29. Mostrar galería de imagenes imágenes.

30. Gestionar grupo.

30.1 Adicionar grupo.

30.2 Modificar grupo.

30.3 Eliminar grupo.

31Mostrar material grupo.

32Gestionar entidad.

32.1 Adicionar entidad.

32.2 Eliminar entidad.

32.3 Modificar entidad.

33Gestionar Evaluacion.

33.1Agregar evaluación.

33.2Modificar evaluación.

33.3Eliminar evaluación.

34Mostrar reporte evaluación de estudiantes.

35Cerrar Sesión.

2.2.2 Requerimientos no funcionales

Requerimientos de Apariencia o Interfaz Externa

La interfaz estará diseñada de modo tal que el usuario pueda tener en todo momento el control de la aplicación, lo que le permitirá ir de un punto a otro dentro de ella con gran facilidad. Se cuidará porque la aplicación sea lo más interactiva posible.

Requerimientos de Usabilidad

El producto está orientado para ser usado por diferentes tipos de usuarios, entre los que se encuentran aquellas personas que tengan poca experiencia en el uso de la computadora y que sólo necesitarían un ligero entrenamiento sobre el funcionamiento de los principales elementos de una interfaz estándar en el ambiente del sistema operativo Windows (uso del Mouse, manejo de menús, botones y cuadros de texto).

Requerimientos de Rendimiento

El sistema propuesto debe ser rápido en el procesamiento de la información así como a la hora de dar respuesta a la solicitud de los usuarios, además debe permitir el acceso simultáneo a los datos por diferentes usuarios. Todo esto depende en gran medida del uso que se le da a los recursos que se disponen en el modelo Cliente/Servidor y de la velocidad de las consultas en la Base de Datos. El sistema deberá recuperarse en un corto período de tiempo ante cualquier falla.

Requerimientos de Soporte

1. Se requiere un servidor de bases de datos con soporte para grandes volúmenes de información, velocidad de procesamiento y tiempo de respuesta rápido en accesos concurrentes. Se documentará la aplicación para garantizar su soporte.
2. El administrador tendrá la responsabilidad de mantener actualizada la aplicación.
3. El sistema deberá permitir futuras mejoras y nuevas opciones que se le quieran incorporar.

Requerimientos de Portabilidad

La plataforma seleccionada para desarrollar la aplicación es Windows, pero puede ser ejecutada desde otras plataformas que soporten el lenguaje PHP, por ejemplo Linux.

Requerimientos de Seguridad

1. No se permitirán accesos sin autorización al sistema, para ello se definirá una política de usuarios con roles y privilegios diferentes para que la información pueda ser consultada por las personas autorizadas. Para ello el sistema contará con cuatro tipos de usuarios, el administrador, el investigador, el director y los usuarios, que según el rol que tengan asignado en el sistema será el acceso a la parte de la información que le corresponde.
2. Debe garantizar la conectividad e integridad de los datos almacenados a través de la red. Esto está garantizado por el Sistema Operativo.
3. Debe garantizar la confidencialidad para proteger la información de acceso no autorizado. Esto estará garantizado por el Sistema Gestor de Base de Datos.

Confiabilidad

Es importante que el sistema presente un mecanismo de respuesta rápida ante fallos y que en caso de ocurrencia se minimicen las pérdidas de información.

Requerimientos de Ayuda y Documentación en Línea

El sistema contará con una ayuda que explicará de manera clara y sencilla al usuario todas las funcionalidades del sistema y esta estará disponible en todas las ventanas del sitio.

Requerimientos de Software

Se debe disponer de un sistema operativo compatible. El sistema debe poderse ejecutar en entornos Windows y/o Linux (Multiplataforma).

Del lado del servidor se utilizará Wamp como servidor Web, PHP 5.3.5 o superior como lenguaje de programación, PostgreSQL como sistema gestor de base de datos para garantizar la integridad y consistencia de la información.

Del lado del cliente se requiere un navegador que interprete las funciones básicas de Java Script, se recomienda Mozilla Firefox 3.0 o superior.

Requerimientos de Hardware

Para el desarrollo y puesta en práctica del proyecto se requieren máquinas con los siguientes requisitos:

1. Procesador de 400MHz y 256 MB de RAM.
2. Requiere 35 MB de espacio de disco duro.
3. Navegador Web.
4. Periférico de comunicación: Módem y tarjeta de red.
5. Las computadoras situadas en los puestos de trabajo de los usuarios requerirán como mínimo:
6. Un Procesador Pentium IV
7. 256 Mb de Memoria RAM
8. Deben estar conectadas en red con el servidor a través de una tarjeta de red de 100 Mbps
9. Teclado y Mouse.

2.3- Descripción del sistema propuesto

2.3.1 Modelo de casos de uso del sistema

El modelo de casos de uso permite que los desarrolladores del software y los clientes lleguen a un acuerdo sobre los requisitos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema. Describe lo que hace el sistema para cada tipo de usuario (Jacobson et al., 2006).

2.3.2 Actores del sistema

Un actor no es más que un rol que juega un usuario de Caso de Uso cuando interaccionan con estos casos de uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez que se han identificado los actores del sistema, se ha identificado el entorno externo del sistema. (Jacobson, Booch, & Rumbaugh, 2006).

Actor del Sistema	Justificación
Profesor	Gestiona la información sobre, la asignatura, es el encargado de todo el proceso soporte y mantenimiento

	del sistema.
Alumno	Consulta y revisa la información.

Tabla 4: Actores del sistema.

2.3.3 Casos de uso del sistema

Dado el número de casos de uso, se introducen paquetes para lograr una mejor organización de los elementos en grupos. Sobre los paquetes del sistema Pressman expresa: Subdividir los casos de uso en paquetes resulta de mucha ayuda en la modelación de cualquier sistema informático. Los paquetes son un mecanismo de organización de elementos que subdividen el modelo en otros más pequeños que colaboran entre sí. (Pressman, 2007).

1. Paquete Administración

1.1 Gestionar usuario

2. Paquete Gestión

2.1 Gestionar clase.

2.2 Gestionar ejercicios.

2.3 Gestionar estudiantes

2.4 Gestionar bibliografía.

3. Paquete Reportes

3.1 Mostrar datos de estudiantes.

3.2 Mostrar evaluación de estudiantes.

4. Paquete Seguridad

4.1 Iniciar sesión

4.2 Cambiar contraseña

4.3 Cerrar sesión

2.3.4 Diagramas de casos de uso del sistema

Relación entre los paquetes

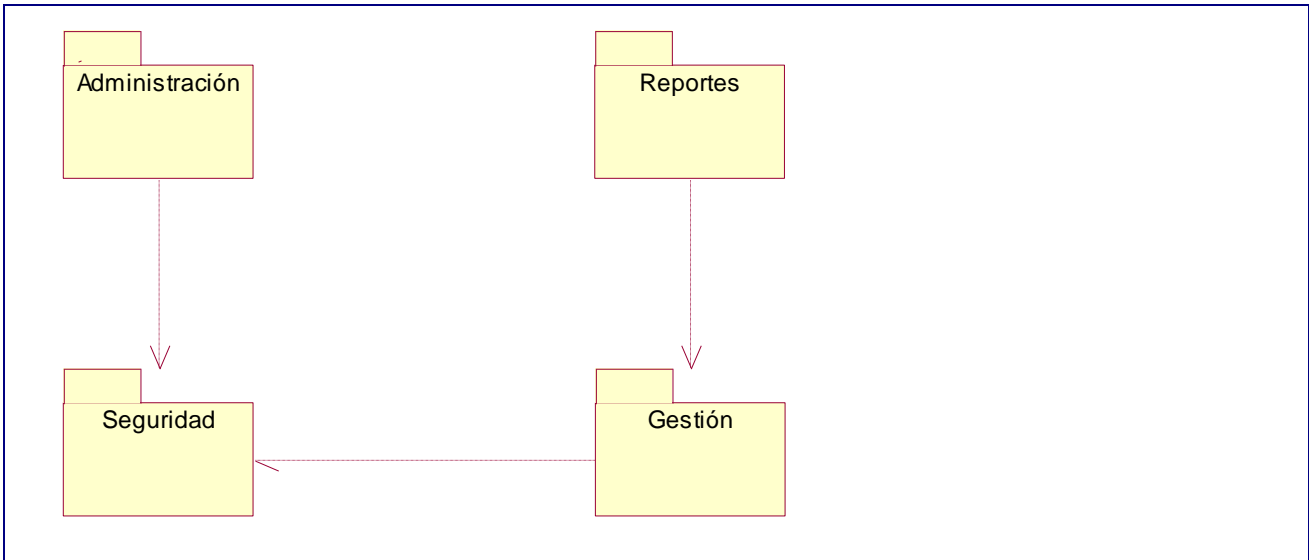


Figura 4: Relación entre los paquetes del sistema.

Requerimientos de Apariencia o Interfaz Externa

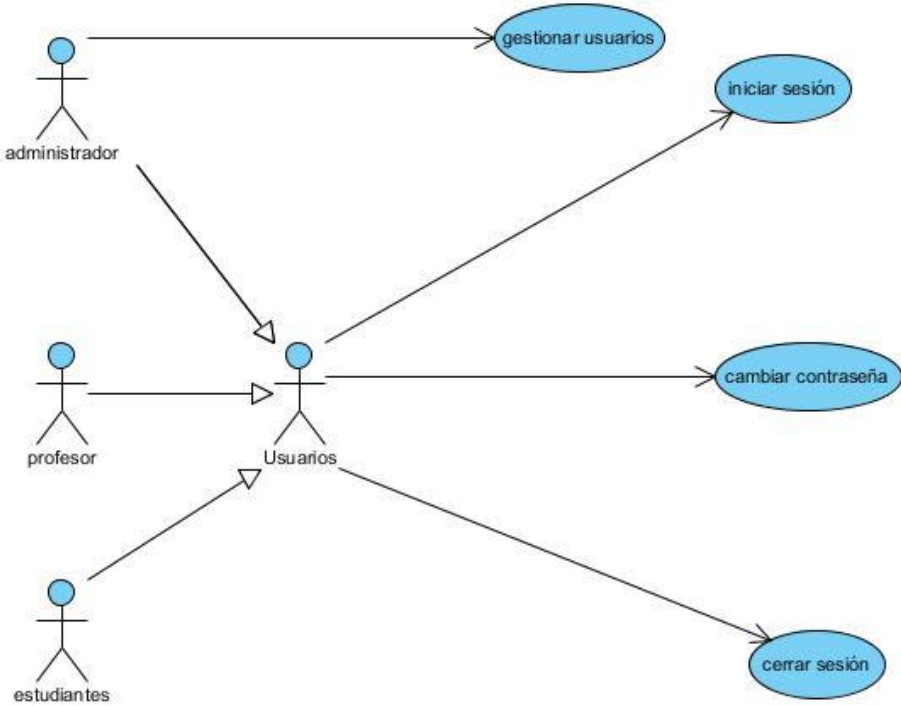


Figura 5. Diagrama de casos de uso. Paquete seguridad.

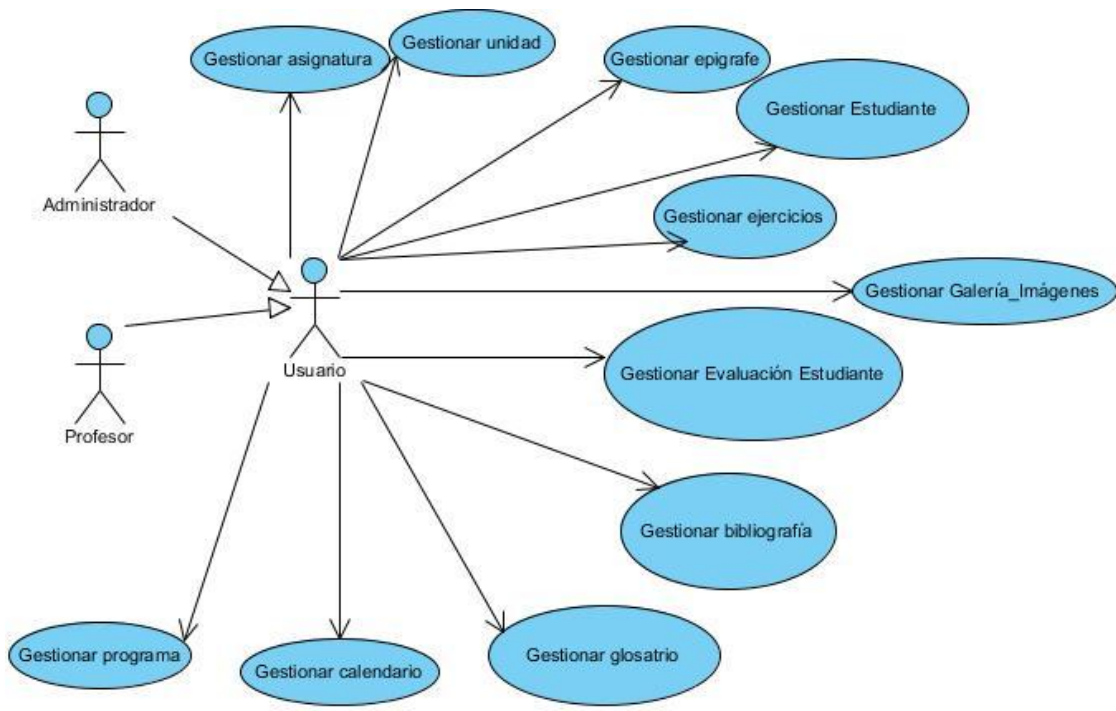


Figura 6. Diagrama de casos de uso. Paquete Gestionar.

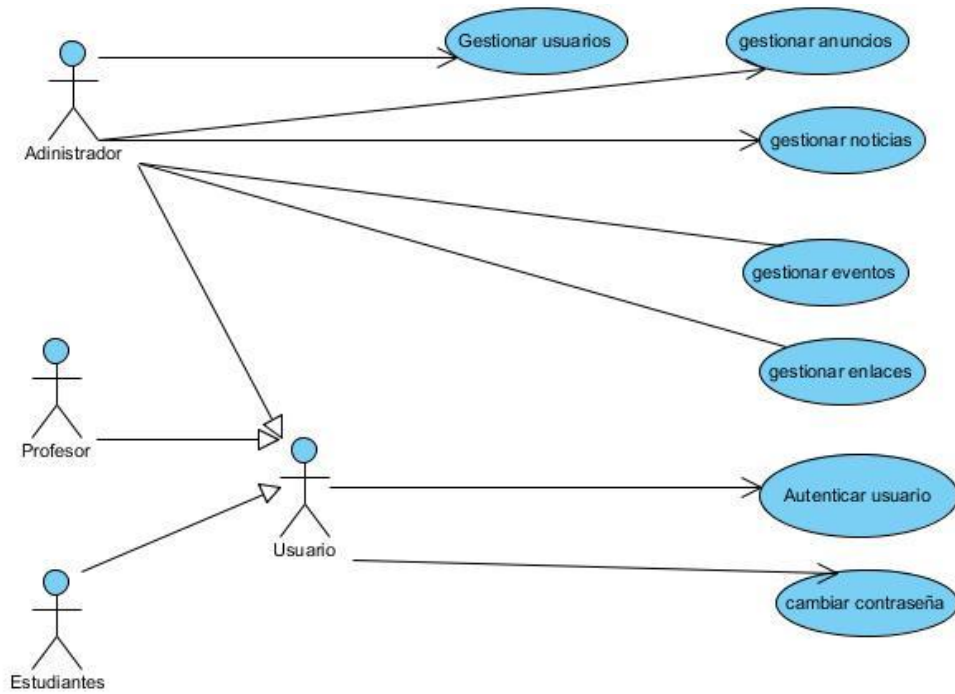


Figura 7. Diagrama de casos de uso. Paquete Administrar.

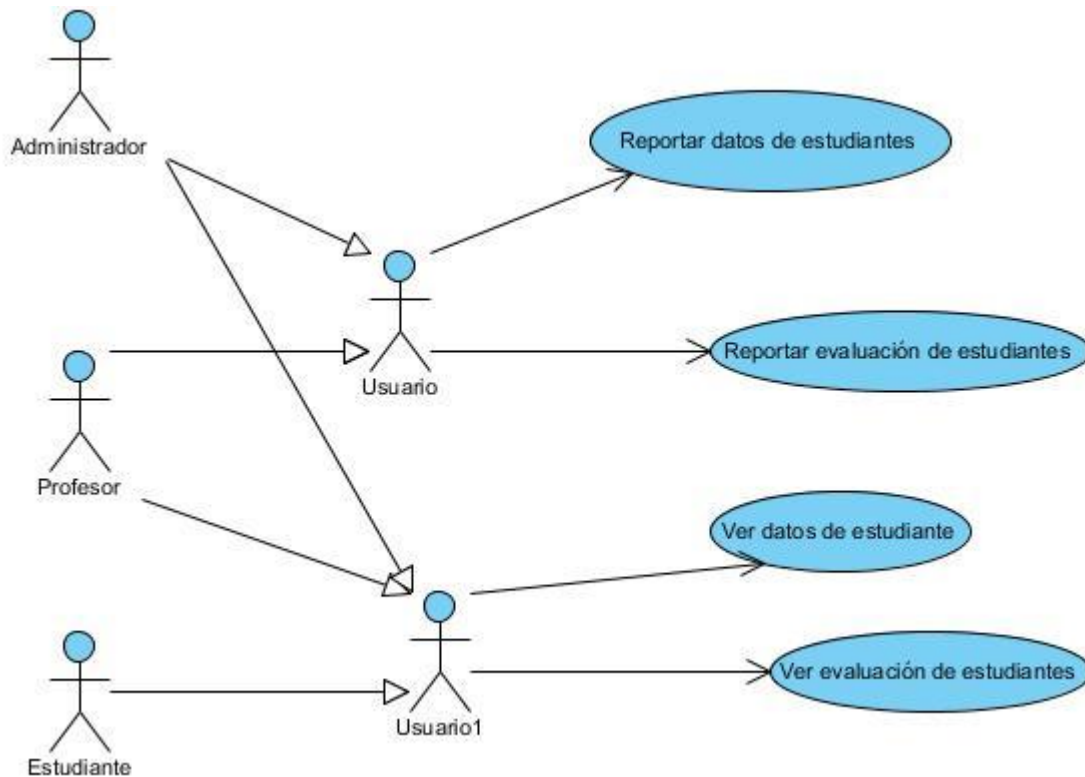


Figura 8. Diagrama de casos de uso. Paquete Reportes.

2.3.5 Descripción de los casos de uso del sistema

Tabla 5. Descripción del caso de uso Crear cuenta.

Caso de uso	Crear cuenta.
Actores	Administrador.
Propósito:	Crear un usuario en el sistema.
Resumen:	El caso de uso inicia cuando el administrador desea crear una nueva cuenta para entrar al sistema. Se muestra el formulario donde introduce los datos a llenar para el nuevo usuario. El caso de uso culmina con la actualización de los datos en el sistema.
Referencias	RF-4
Prototipo	Anexo 2

Tabla 6. Descripción del caso de uso Autenticar usuario.

Caso de uso	Autenticar usuario.
Actores	Estudiante, Profesor, Administrador.
Propósito:	Entrar al sistema.

Resumen: El caso de uso inicia cuando el usuario desea entrar al sistema. Se muestra el formulario donde introduce su nombre y su contraseña. El caso de uso culmina con la entrada de los usuarios al sistema.	
Referencias	RF-1, RF-1.1, RF-2.1,RF-2.2
Prototipo	Anexo 3

Tabla 7. Descripción del caso de uso Cambiar contraseña.

Caso de uso	Cambiar contraseña.
Actores	Estudiante, Profesor, Administrador.
Propósito:	Cambiar la contraseña.
Resumen: El caso de uso inicia cuando el usuario desea cambiar su contraseña. Se muestra un formulario donde introduce su actual y nueva contraseña. El caso de uso culmina con la actualización de los datos en el sistema.	
Referencias	RF-3
Prototipo	Anexo 4

Tabla 8. Descripción del caso de uso Gestionar usuario.

Caso de uso	Gestionar usuario.
Actores	Administrador.
Propósito:	Adicionar, modificar o eliminar un usuario al sistema.
Resumen: El caso de uso inicia por tres vías diferentes: <ul style="list-style-type: none"> ✓ Cuando el administrador desea agregar un usuario al sistema. Se muestra el formulario donde este agrega el usuario y sus datos. ✓ Cuando el administrador desea modificar un usuario existente. Se muestran los usuarios registrados en el sistema, selecciona el que desea modificar y realiza los cambios. ✓ Cuando el administrador desea eliminar un usuario del sistema. Se muestran los usuarios registrados en el sistema, selecciona el que desea eliminar y realiza los cambios. El caso de uso culmina con la actualización de los datos en el sistema.	
Referencias	RF-4, RF-4.1, RF-4.2, RF-4.3

Prototipo	Anexo 5
------------------	---------

Tabla 9. Descripción del caso de uso Gestionar Asignatura.

Caso de uso	Gestionar Asignatura.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar una asignatura.
<p>Resumen: El caso de uso inicia por tres vías diferentes:</p> <ul style="list-style-type: none"> ✓ Cuando el usuario desea agregar una asignatura al sistema. Se muestra el área de texto donde este agrega la asignatura. ✓ Cuando el usuario desea modificar una asignatura existente. Se muestra un listado de las asignaturas existentes, selecciona la que desea modificar y realiza los cambios. ✓ Cuando el usuario desea eliminar una asignatura al sistema. Se muestra un listado de las asignaturas existentes, selecciona la que desea eliminar y realiza los cambios. <p>El caso de uso culmina con la actualización de los datos en el sistema.</p>	

Tabla 10. Descripción del caso de uso Gestionar Estudiantes.

Caso de uso	Gestionar Estudiantes.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar un estudiante.
<p>Resumen: El caso de uso inicia por tres vías diferentes:</p> <ul style="list-style-type: none"> ✓ Cuando el usuario desea agregar un <i>estudiante</i>. al sistema. Se muestra el área de texto donde este agrega el <i>estudiante</i>. ✓ Cuando el usuario desea modificar un estudiante. existente. Se muestra un listado de los estudiantes. existentes, selecciona el que desea modificar y realiza los cambios. ✓ Cuando el usuario desea eliminar un estudiante al sistema. Se muestra un listado del estudiante existente, selecciona el que desea eliminar y realiza los cambios. <p>El caso de uso culmina con la actualización de los datos en el sistema.</p>	

Tabla 11. Descripción del caso de uso Gestionar Estudiantes.

Caso de uso	Gestionar Evaluaciones.
--------------------	-------------------------

Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar una evaluación.
<p>Resumen: El caso de uso inicia por tres vías diferentes:</p> <ul style="list-style-type: none"> ✓ Cuando el usuario desea agregar una evaluación. al sistema. Se muestra el área de texto donde este agrega la evaluación. ✓ Cuando el usuario desea modificar una evaluación. existente. Se muestra un listado de las evaluaciones. existentes, selecciona el que desea modificar y realiza los cambios. ✓ Cuando el usuario desea eliminar una evaluación al sistema. Se muestra un listado de las evaluaciones existentes, selecciona la que desea eliminar y realiza los cambios. <p>El caso de uso culmina con la actualización de los datos en el sistema.</p>	

Tabla 12. Descripción del caso de uso Gestionar programas.

Caso de uso	Gestionar programa.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar un Programa.
<p>Resumen: El caso de uso inicia por tres vías diferentes:</p> <ul style="list-style-type: none"> ✓ Cuando el usuario desea agregar un programa al sistema. Se muestra el área de texto donde este agrega el programa. ✓ Cuando el usuario desea modificar un programa existente. Se muestra un listado de los programas existentes, selecciona el que desea modificar y realiza los cambios. ✓ Cuando el usuario desea eliminar un programa existente del sistema. Se muestra un listado de los programas existentes, selecciona el que desea eliminar y realiza los cambios. <p>El caso de uso culmina con la actualización de los datos en el sistema.</p>	

Tabla 13. Descripción del caso de uso Gestionar unidad.

Caso de uso	Gestionar unidad.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar una unidad en el sistema.

Resumen: El caso de uso inicia por tres vías diferentes:

- ✓ Cuando el usuario desea agregar una unidad al sistema. Se muestra el área de texto donde este agrega la unidad.
- ✓ Cuando el usuario desea modificar una unidad existente. Se muestra un listado de las unidades existentes, selecciona la que desea modificar y realiza los cambios.
- ✓ Cuando el usuario desea eliminar una unidad al sistema. Se muestra un listado de las unidades existentes, selecciona el que desea eliminar y realiza los cambios.

El caso de uso culmina con la actualización de los datos en el sistema.

Tabla 14. Descripción del caso de uso Gestionar epígrafe.

Caso de uso	Gestionar epígrafe.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar un epígrafe en el sistema.

Resumen: El caso de uso inicia por tres vías diferentes:

- ✓ Cuando el usuario desea agregar un epígrafe al sistema. Se muestra el área de texto donde este agrega el epígrafe.
- ✓ Cuando el usuario desea modificar un epígrafe existente. Se muestra un listado de los epígrafes existentes, selecciona el que desea modificar y realiza los cambios.
- ✓ Cuando el usuario desea eliminar un epígrafe al sistema. Se muestra un listado de los epígrafes existentes, selecciona el que desea eliminar y realiza los cambios.

El caso de uso culmina con la actualización de los datos en el sistema.

Tabla 15. Descripción del caso de uso Gestionar bibliografía.

Caso de uso	Gestionar bibliografía.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar una bibliografía.

Resumen: El caso de uso inicia por tres vías diferentes:

- ✓ Cuando el usuario desea agregar una bibliografía al sistema. Se muestra el área de texto donde este agrega la bibliografía.
- ✓ Cuando el usuario desea modificar una bibliografía existente. Se muestra un listado de las bibliografías existentes, selecciona la que desea modificar y realiza los cambios.
- ✓ Cuando el usuario desea eliminar una bibliografía al sistema. Se muestra un

listado de las bibliografías existentes, selecciona la que desea eliminar y realiza los cambios.

El caso de uso culmina con la actualización de los datos en el sistema.

Tabla 16. Descripción del caso de uso Gestionar glosario.

Caso de uso	Gestionar glosario.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar un glosario en el sistema.
<p>Resumen: El caso de uso inicia por tres vías diferentes:</p> <ul style="list-style-type: none"> ✓ Cuando el usuario desea agregar un glosario al sistema. Se muestra el área de texto donde este agrega el glosario. ✓ Cuando el usuario desea modificar un glosario existente. Se muestra un listado de los glosarios existentes, selecciona el que desea modificar y realiza los cambios. ✓ Cuando el usuario desea eliminar un glosario al sistema. Se muestra un listado de los glosarios existentes, selecciona el que desea eliminar y realiza los cambios. <p>El caso de uso culmina con la actualización de los datos en el sistema.</p>	

Tabla 17. Descripción del caso de uso Gestionar ejercicios.

Caso de uso	Gestionar ejercicios.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar un ejercicio en el sistema.
<p>Resumen: El caso de uso inicia por tres vías diferentes:</p> <ul style="list-style-type: none"> ✓ Cuando el usuario desea agregar un ejercicio propuesto al sistema. Se muestra el área de texto donde este agrega el ejercicio. ✓ Cuando el usuario desea modificar un ejercicio propuesto existente. Se muestra un listado de los ejercicios existentes, selecciona el que desea modificar y realiza los cambios. ✓ Cuando el usuario desea eliminar un ejercicio propuesto del sistema. Se muestra un listado de los ejercicios propuestos, selecciona el que desea eliminar y realiza los cambios. <p>El caso de uso culmina con la actualización de los datos en el sistema.</p>	

Tabla 18. Descripción del caso de uso Gestionar entidad.

Caso de uso	Gestionar entidad.
Actores	Administrador.
Propósito:	Adicionar, modificar o eliminar una entidad en el sistema.
<p>Resumen: El caso de uso inicia por tres vías diferentes:</p> <ul style="list-style-type: none"> ✓ Cuando el usuario desea agregar una entidad al sistema. Se muestra el área de texto donde este agrega la entidad. ✓ Cuando el usuario desea modificar una entidad existente. Se muestra un listado de las entidades existentes, selecciona la que desea modificar y realiza los cambios. ✓ Cuando el usuario desea eliminar una entidad del sistema. Se muestra un listado de las entidades existentes, selecciona el que desea eliminar y realiza los cambios. <p>El caso de uso culmina con la actualización de los datos en el sistema.</p>	

Tabla 19. Descripción del caso de uso Gestionar grupo.

Caso de uso	Gestionar grupo.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar un grupo en el sistema.
<p>Resumen: El caso de uso inicia por tres vías diferentes:</p> <ul style="list-style-type: none"> ✓ Cuando el usuario desea agregar un grupo al sistema. Se muestra el área de texto donde este agrega el grupo. ✓ Cuando el usuario desea modificar un grupo existente. Se muestra un listado de los grupos existentes, selecciona el que desea modificar y realiza los cambios. ✓ Cuando el usuario desea eliminar un grupo del sistema. Se muestra un listado de los grupos existentes, selecciona el que desea eliminar y realiza los cambios. <p>El caso de uso culmina con la actualización de los datos en el sistema.</p>	

Tabla 20. Descripción del caso de uso Gestionar calendario.

Caso de uso	Gestionar calendario.
Actores	Profesor, Administrador.
Propósito:	Adicionar, modificar o eliminar un calendario.

Resumen: El caso de uso inicia por tres vías diferentes:

- ✓ Cuando el usuario desea agregar un calendario al sistema. Se muestra el área de texto donde este agrega el calendario.
- ✓ Cuando el usuario desea modificar un calendario existente. Se muestra un listado de los calendarios existentes, selecciona el que desea modificar y realiza los cambios.
- ✓ Cuando el usuario desea eliminar un calendario del sistema. Se muestra un listado de los calendarios existentes, selecciona el que desea eliminar y realiza los cambios.

El caso de uso culmina con la actualización de los datos en el sistema.

Tabla 21. Descripción del caso de uso Gestionar anuncios.

Caso de uso	Gestionar anuncios.
Actores	Administrador.
Propósito:	Adicionar, modificar o eliminar un anuncio.

Resumen: El caso de uso inicia por tres vías diferentes:

- ✓ Cuando el usuario desea agregar un anuncios al sistema. Se muestra el área de texto donde este agrega el anuncio.
- ✓ Cuando el usuario desea modificar un anuncios existente. Se muestra un listado de los anuncios existentes, selecciona el que desea modificar y realiza los cambios.
- ✓ Cuando el usuario desea eliminar un anuncio del sistema. Se muestra un listado de los anuncios existentes, selecciona el que desea eliminar y realiza los cambios.

El caso de uso culmina con la actualización de los datos en el sistema.

Tabla 22. Descripción del caso de uso Gestionar noticias.

Caso de uso	Gestionar noticias.
Actores	Administrador.
Propósito:	Adicionar, modificar o eliminar una noticia.

Resumen: El caso de uso inicia por tres vías diferentes:

- ✓ Cuando el usuario desea agregar una noticia al sistema. Se muestra el área de texto donde este agrega la noticia.

- ✓ Cuando el usuario desea modificar una noticia existente. Se muestra un listado de las noticias existentes, selecciona la que desea modificar y realiza los cambios.
- ✓ Cuando el usuario desea eliminar una noticia del sistema. Se muestra un listado de las noticias existentes, selecciona la que desea eliminar y realiza los cambios.

El caso de uso culmina con la actualización de los datos en el sistema.

Tabla 23. Descripción del caso de uso Gestionar evento.

Caso de uso	Gestionar evento.
Actores	Administrador.
Propósito:	Adicionar, modificar o eliminar un evento.
<p>Resumen: El caso de uso inicia por tres vías diferentes:</p> <ul style="list-style-type: none"> ✓ Cuando el usuario desea agregar un evento al sistema. Se muestra el área de texto donde este agrega el evento. ✓ Cuando el usuario desea modificar un evento existente. Se muestra un listado de los eventos existentes, selecciona el que desea modificar y realiza los cambios. ✓ Cuando el usuario desea eliminar un evento del sistema. Se muestra un listado de los eventos existentes, selecciona el que desea eliminar y realiza los cambios. <p>El caso de uso culmina con la actualización de los datos en el sistema.</p>	

Conclusiones Parciales del capítulo

En este capítulo se realizó un análisis para conocer cómo funciona el negocio en el proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura de octavo grado de la enseñanza preuniversitaria en Cuba, se estudiaron los procesos que lo conforman, lo que permitió modelarlo mediante reglas, casos de uso, actores y trabajadores que en él participan. Se definieron los principales requerimientos funcionales del sistema, así como los actores y su relación con estos, se definieron los paquetes de casos de uso que permiten agruparlos según su funcionalidad y por último se realiza una descripción de cada caso de uso.

Capítulo 3: Implementación de la aplicación propuesta para apoyar el proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura de onceno grado en la enseñanza preuniversitaria en Cuba:

Introducción

El presente capítulo abarca todos los aspectos referentes al diseño de la solución propuesta. Se presentará el diagrama de componente y de despliegue, que resultaron del diseño realizado de cada uno de los casos de uso del sistema planteados. Se planteará también el modelo de datos a partir del diagrama de clases persistentes. Por último, se hará referencia a los estándares de diseño y de programación que se tienen en cuenta.

3.1 Diagrama de Clases Web

El diagrama de clases es el principal diagrama de análisis y diseño, en él se representan las clases, con sus atributos, operaciones, interfaces y relaciones que existen entre ellas; se puede dividir en capas, aplicación y datos, las cuales muestran por separado las clases que intervienen en la lógica de la aplicación, la interfaz de usuario y el almacenamiento de datos. El diagrama de clases Web, representa a las páginas lógicas como clase, y de igual manera se representan sus relaciones, sobre el tema Jacobson, Booch & Rumbaugh expresan: “Un diagrama de clases Web representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase. Al tratar de utilizar el diagrama de clases tradicional para modelar aplicaciones Web surgen varios problemas, por lo cual los especialistas del Rational plantearon la creación de una extensión al modelo de análisis y diseño que permitiera representar el nivel de abstracción adecuado y la relación con los restantes artefactos de UML” (Jacobson, Booch, & Rumbaugh, 2006).

Teniendo en cuenta lo anterior y haciendo uso de las extensiones de UML para Web se elaboraron los siguientes diagramas de clases Web:

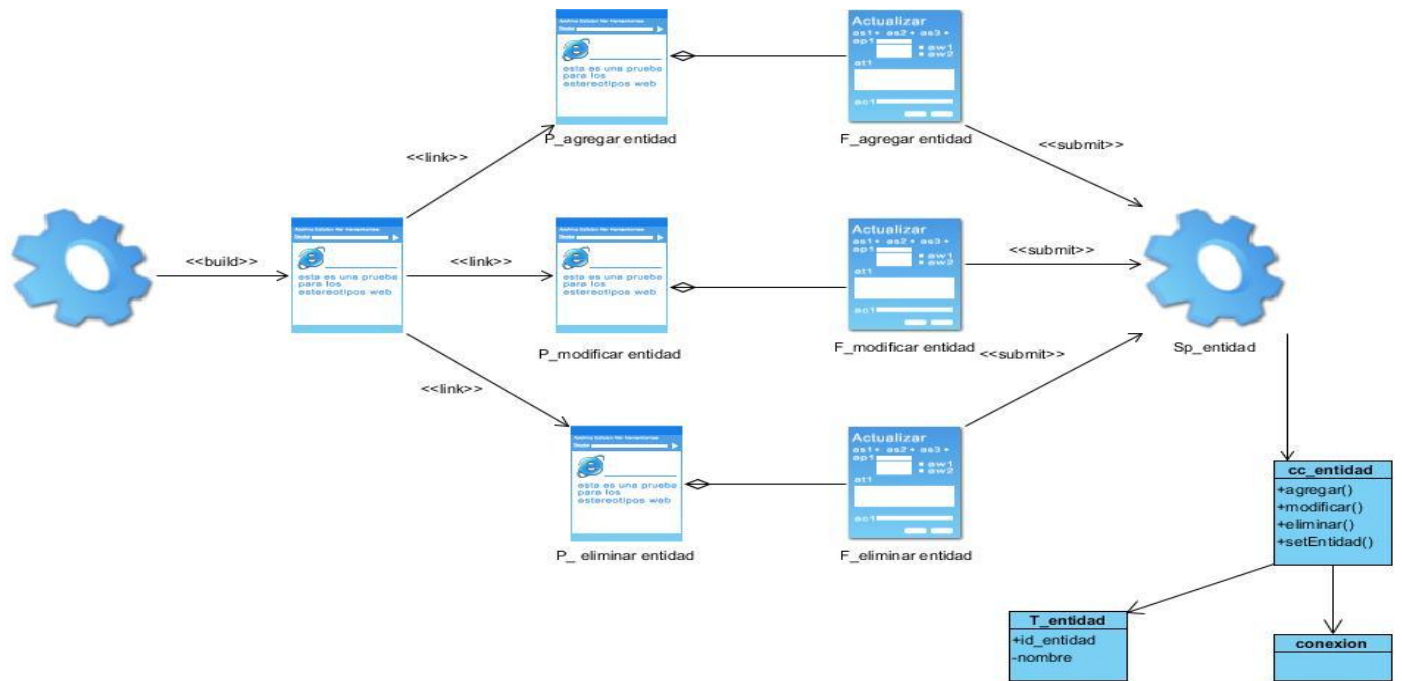


Figura 9. Diagramas de clases de diseño. Caso de uso Gestionar entidad.

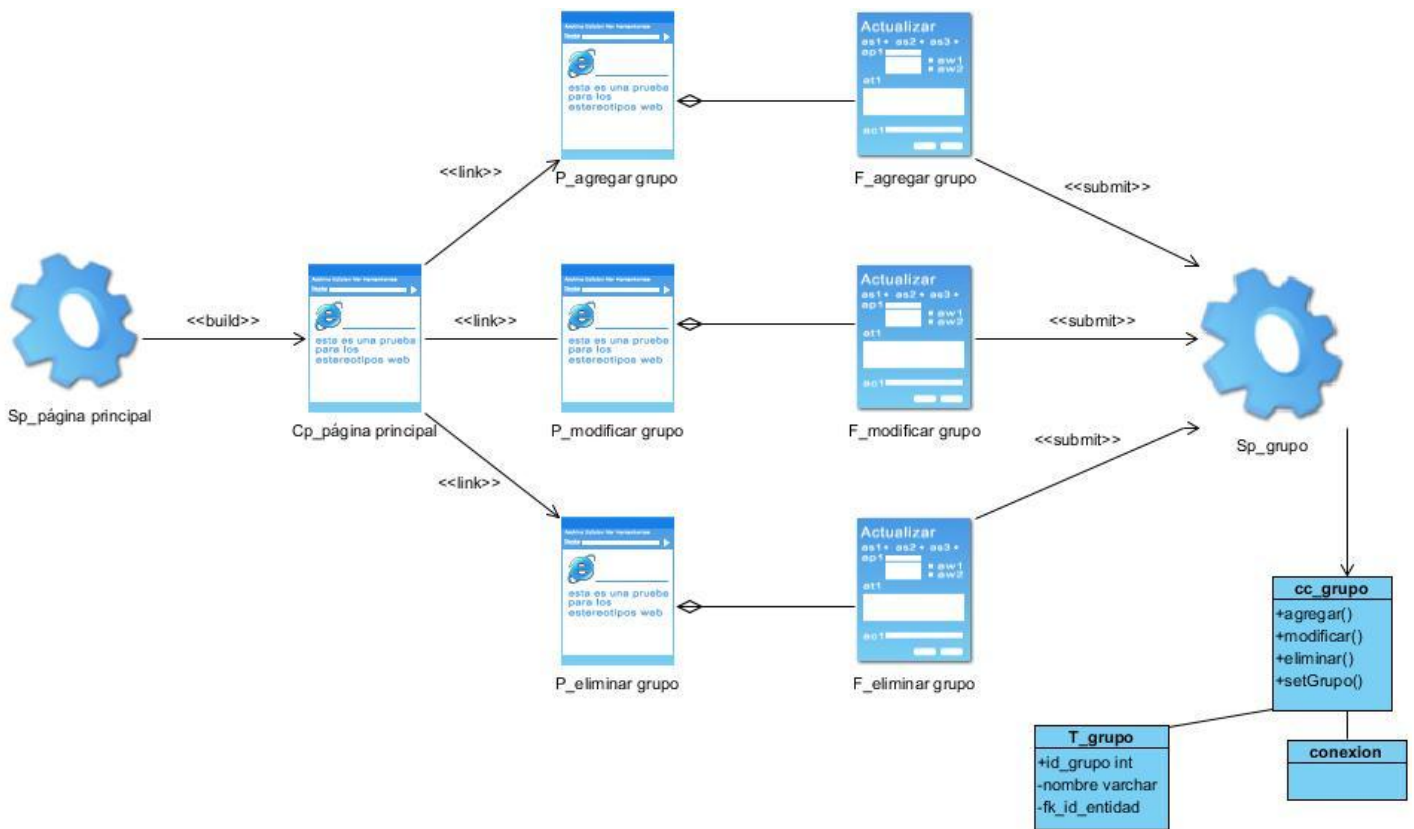


Figura 10. Diagramas de clases de diseño. Caso de uso Gestionar grupo.

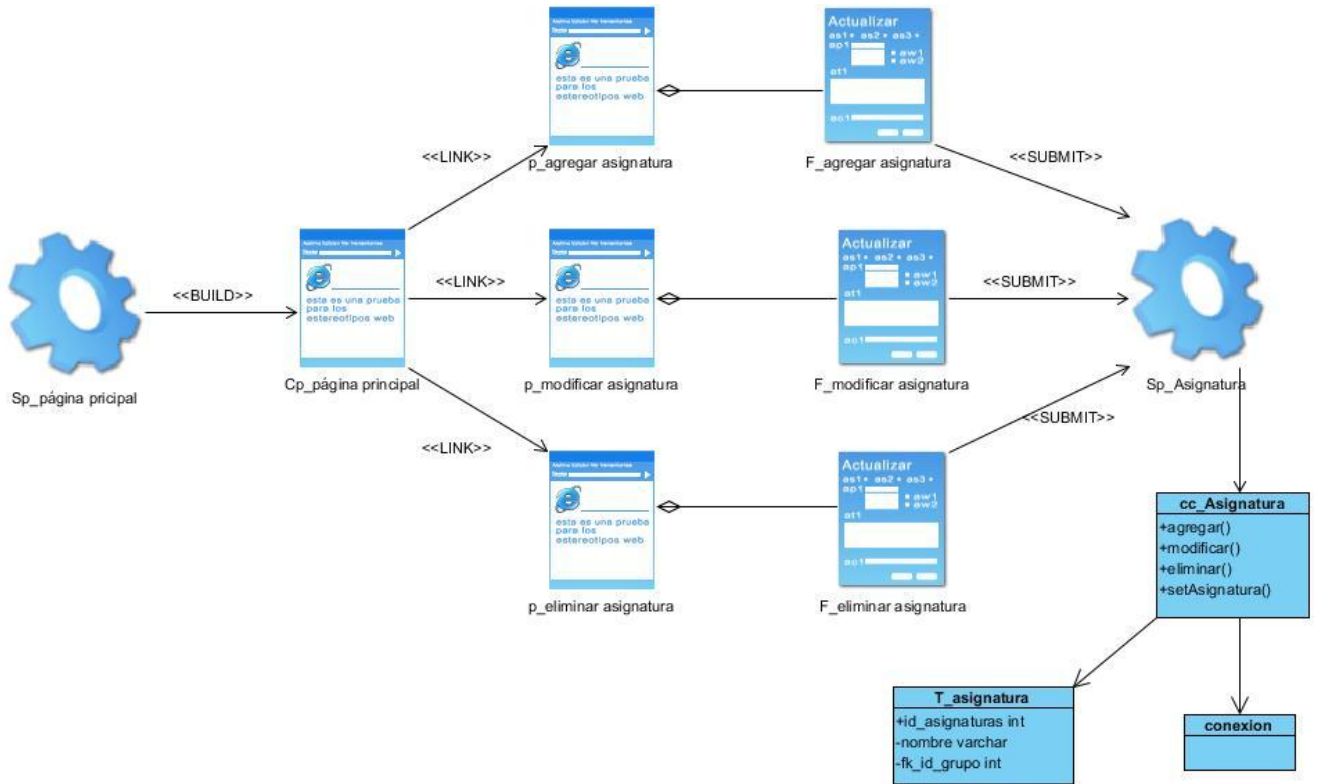


Figura 11. Diagramas de clases de diseño. Caso de uso Gestionar asignaturas.

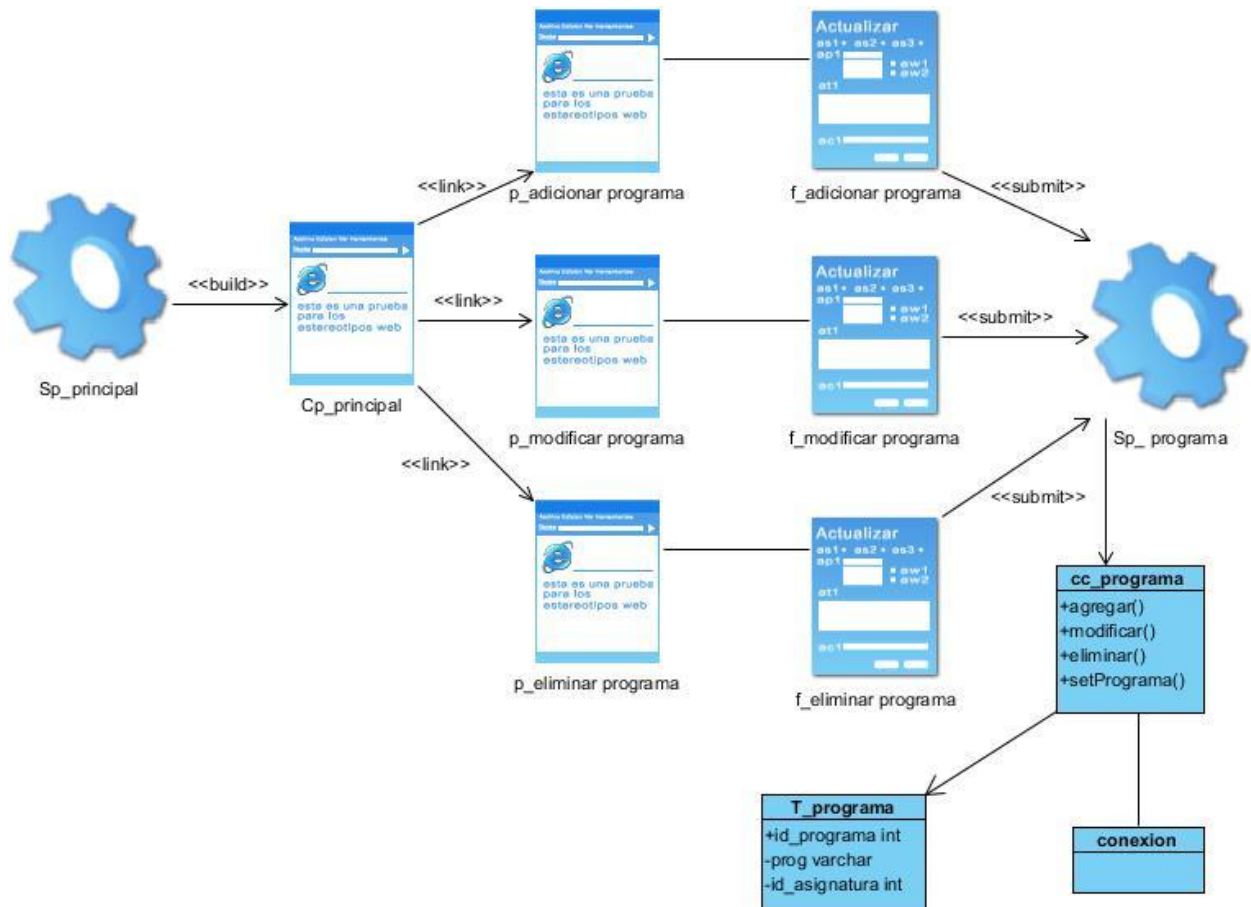


Figura 12. Diagramas de clases de diseño. Caso de uso Gestionar programa.

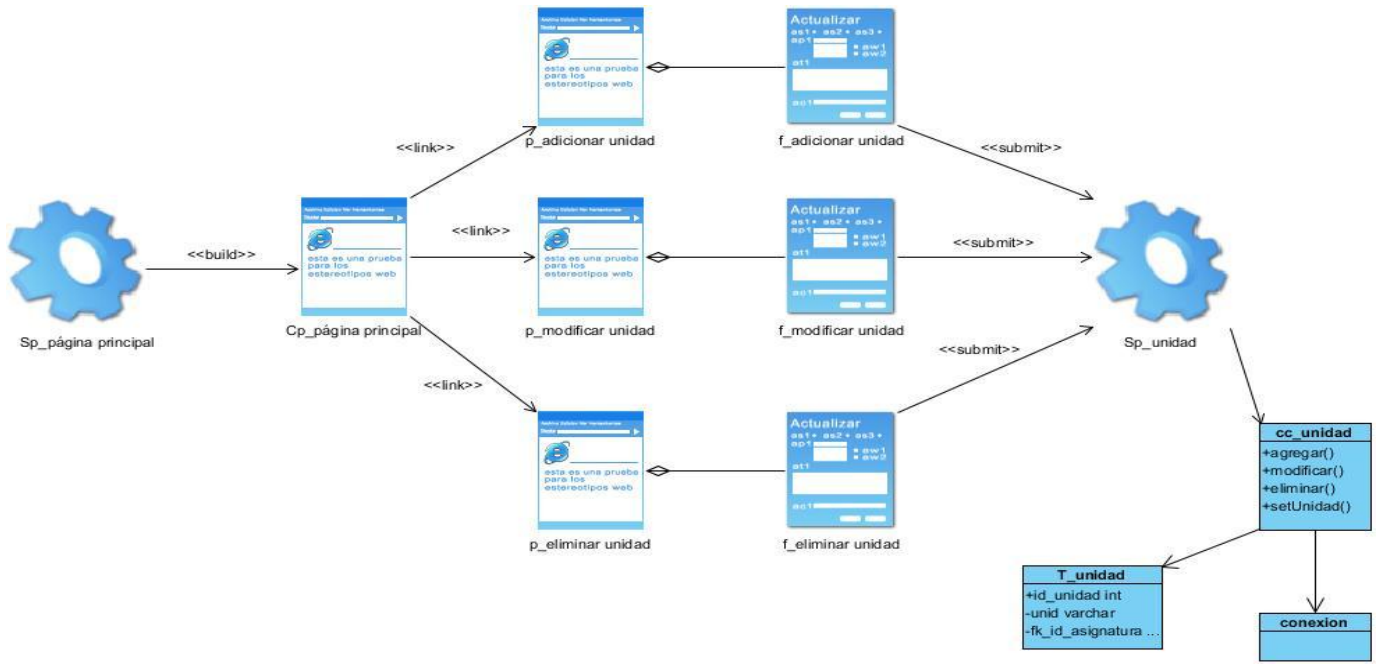


Figura 13. Diagramas de clases de diseño. Caso de uso Gestionar unidad.

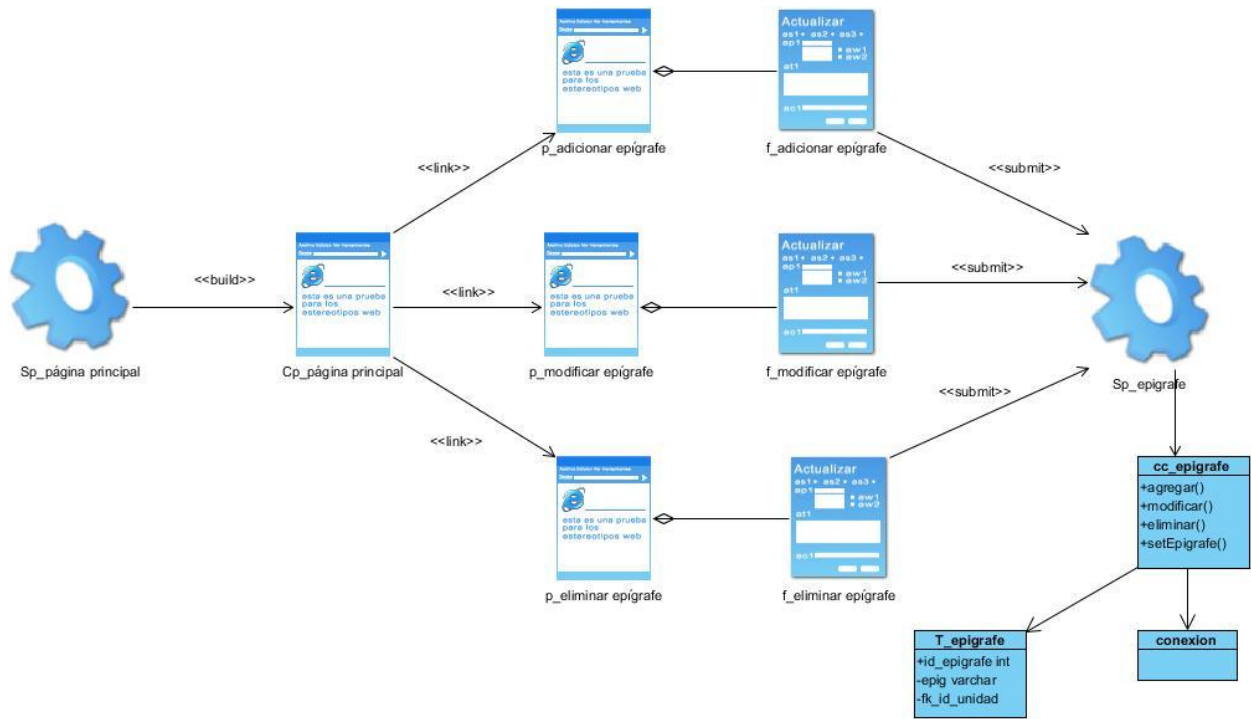


Figura 14. Diagramas de clases de diseño. Caso de uso Gestionar epígrafe.

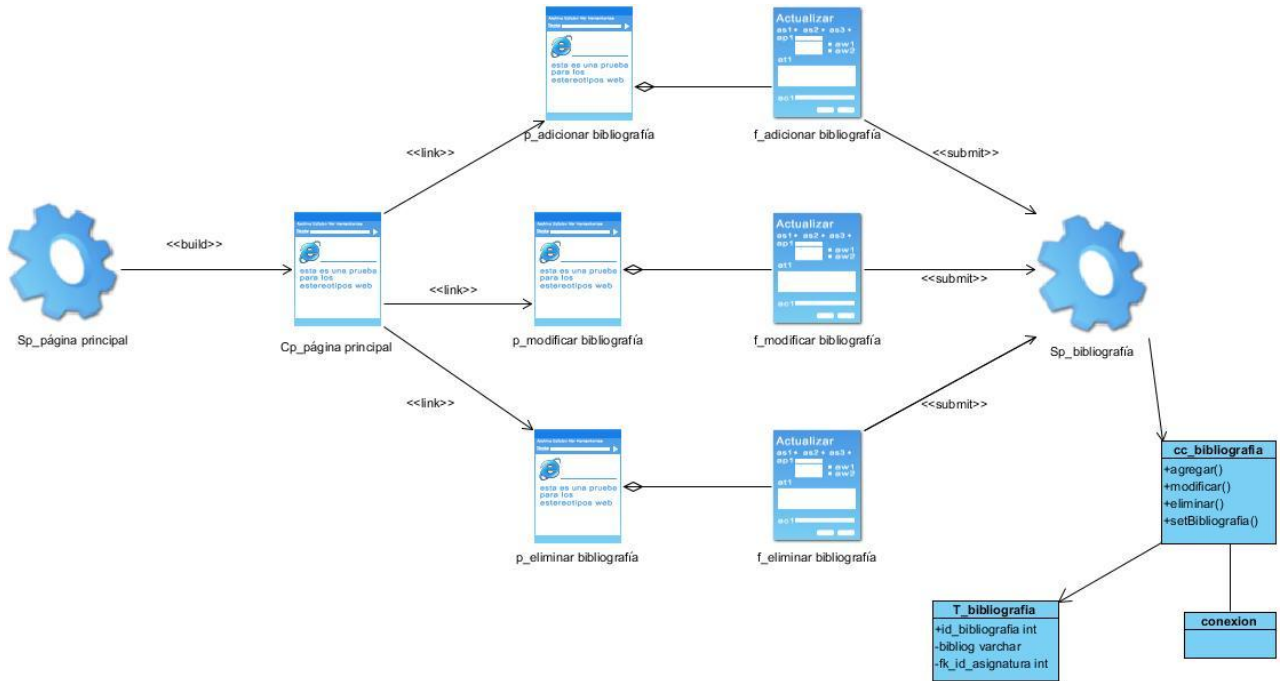


Figura 15. Diagramas de clases de diseño. Caso de uso Gestionar bibliografía.

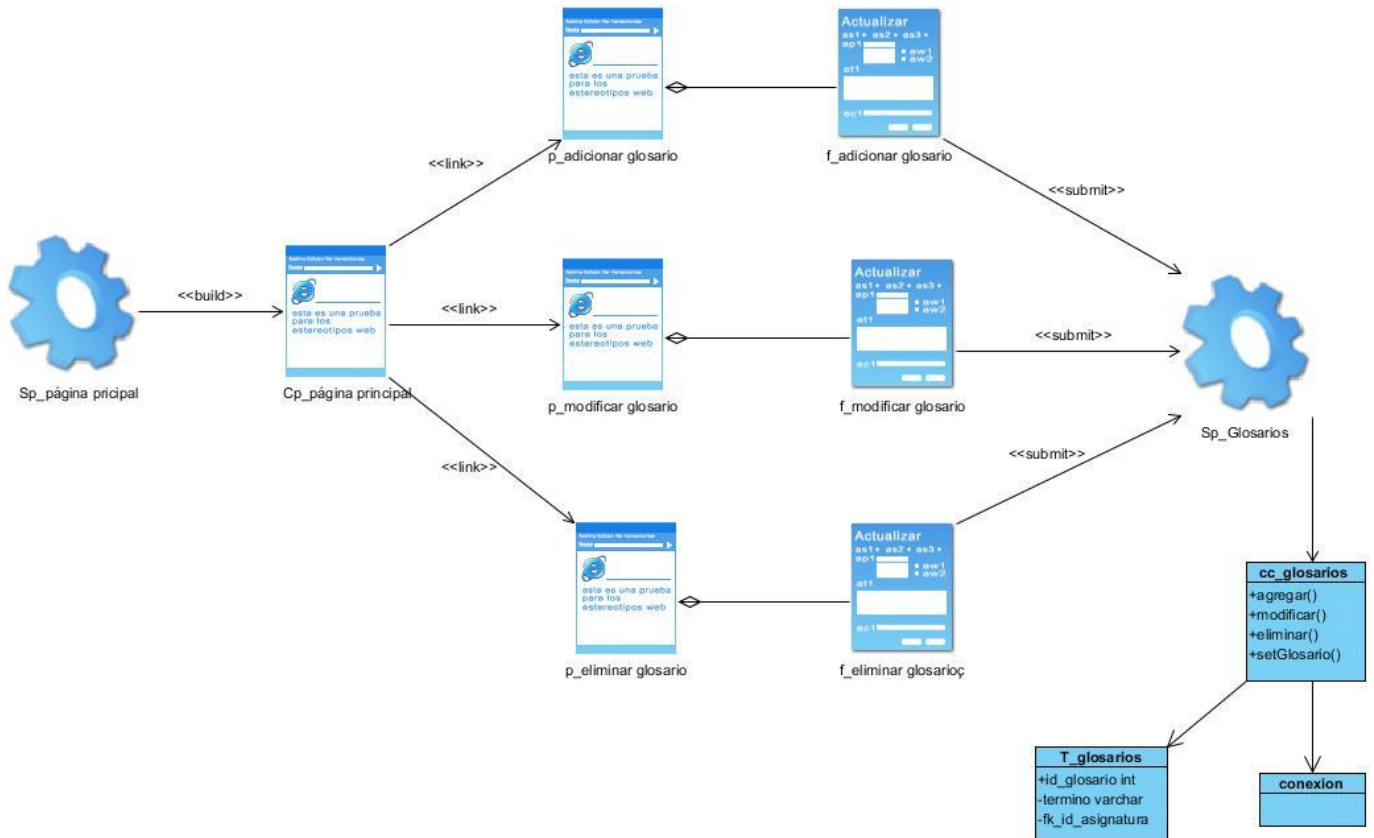


Figura 16. Diagramas de clases de diseño. Caso de uso Gestionar glosario de términos.

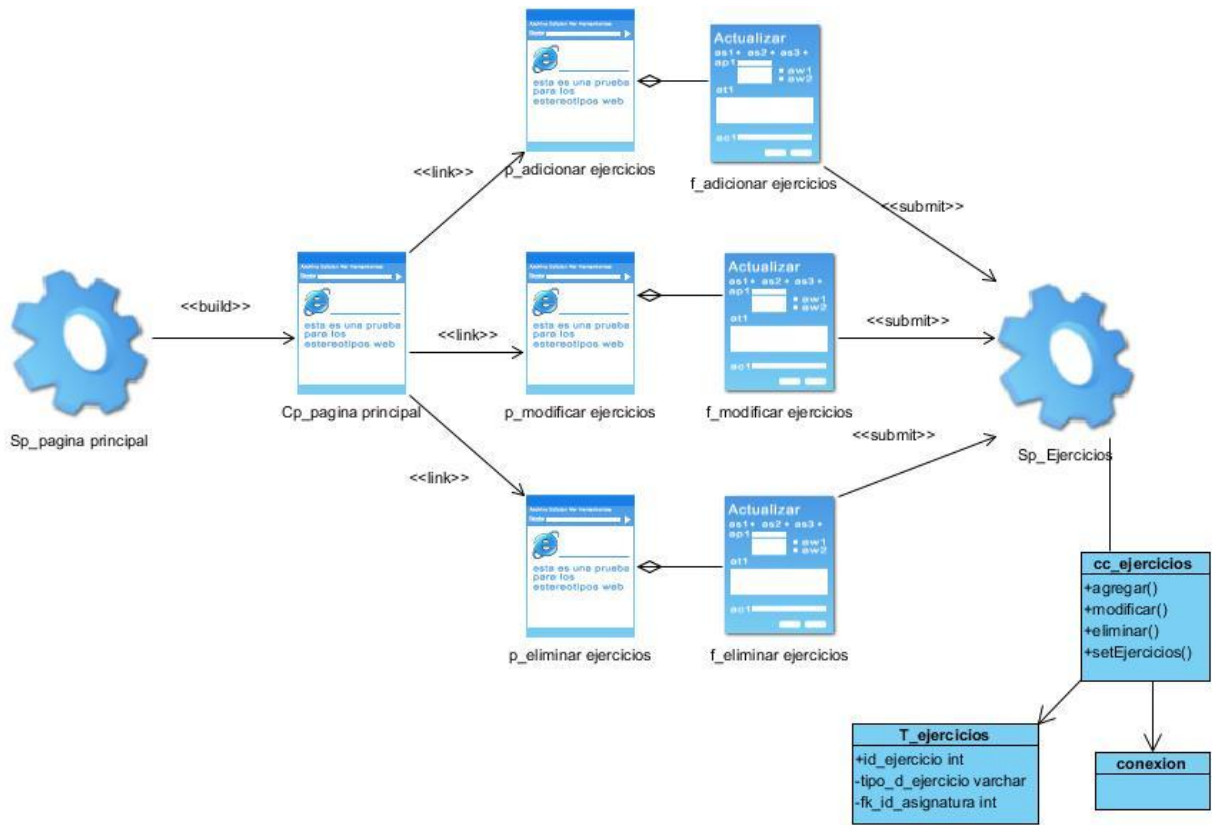


Figura 17. Diagramas de clases de diseño. Caso de uso Gestionar ejercicios.

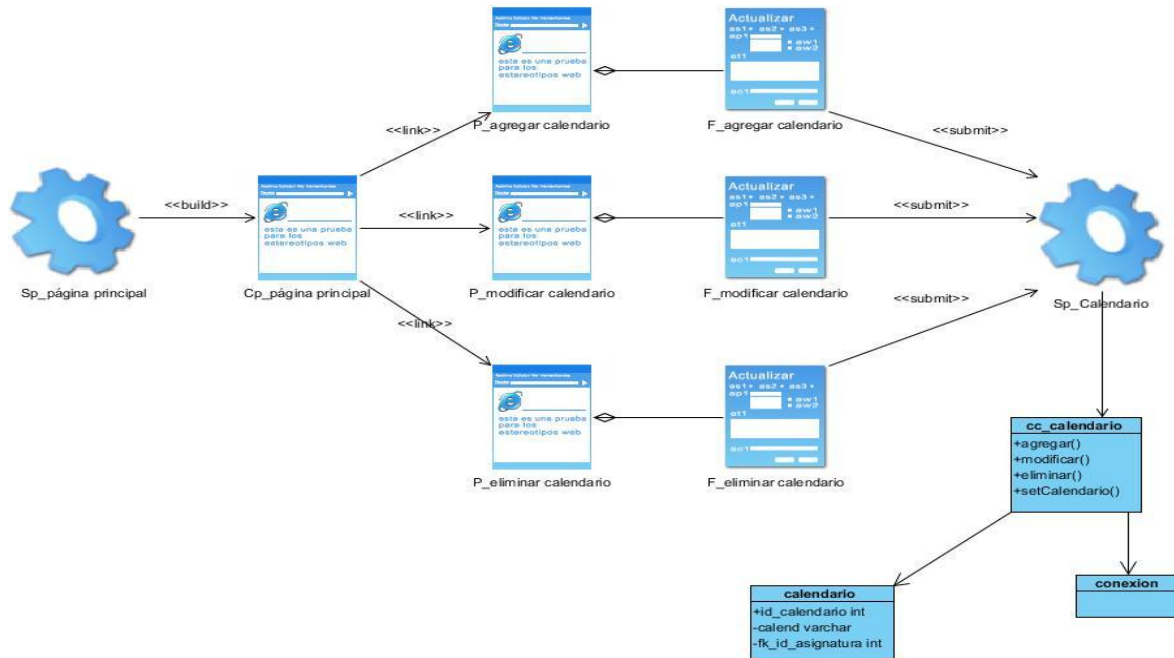


Figura 18. Diagramas de clases de diseño. Caso de uso Gestionar calendario.

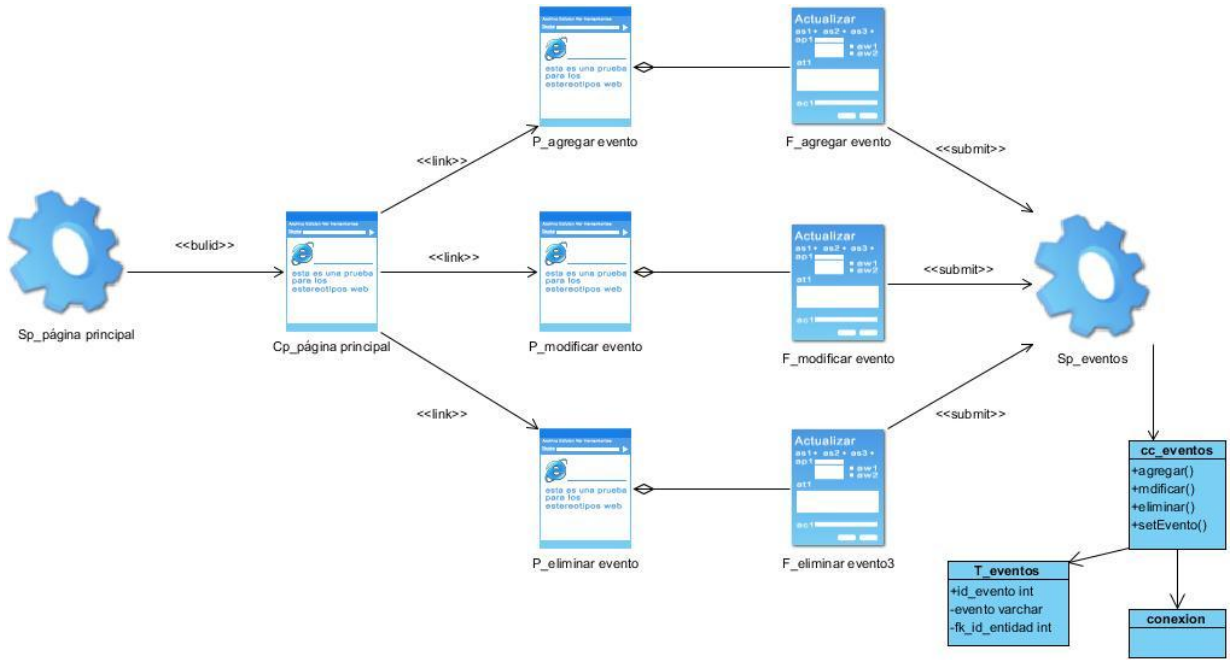


Figura 19. Diagramas de clases de diseño. Caso de uso Gestionar eventos.

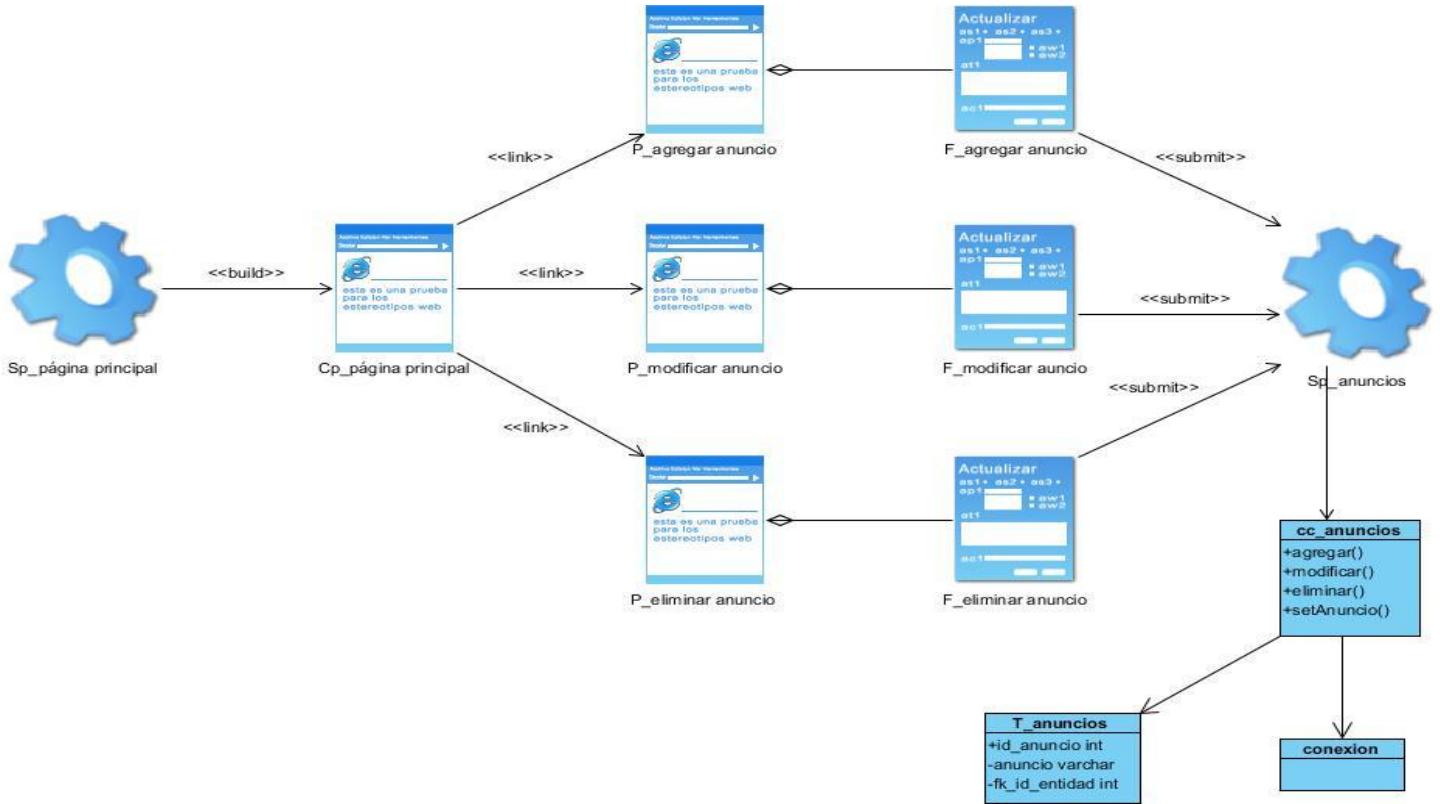


Figura 20. Diagramas de clases de diseño. Caso de uso Gestionar anuncios.

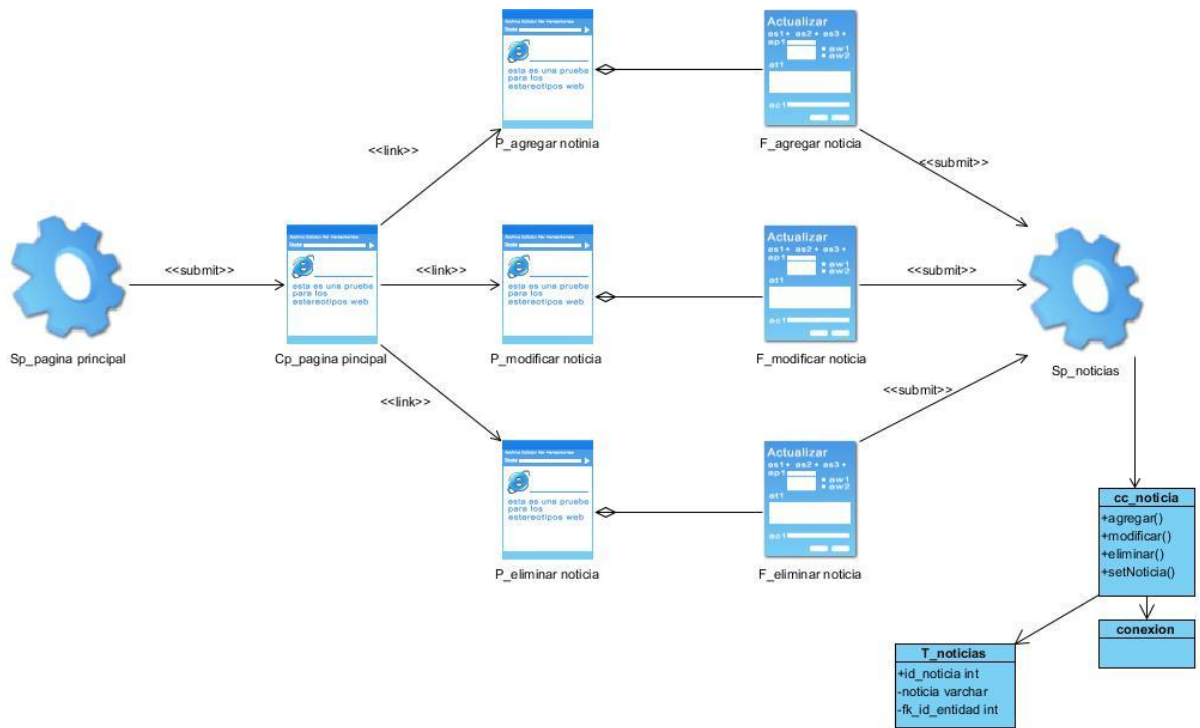


Figura 21. Diagramas de clases de diseño. Caso de uso Gestionar noticias.

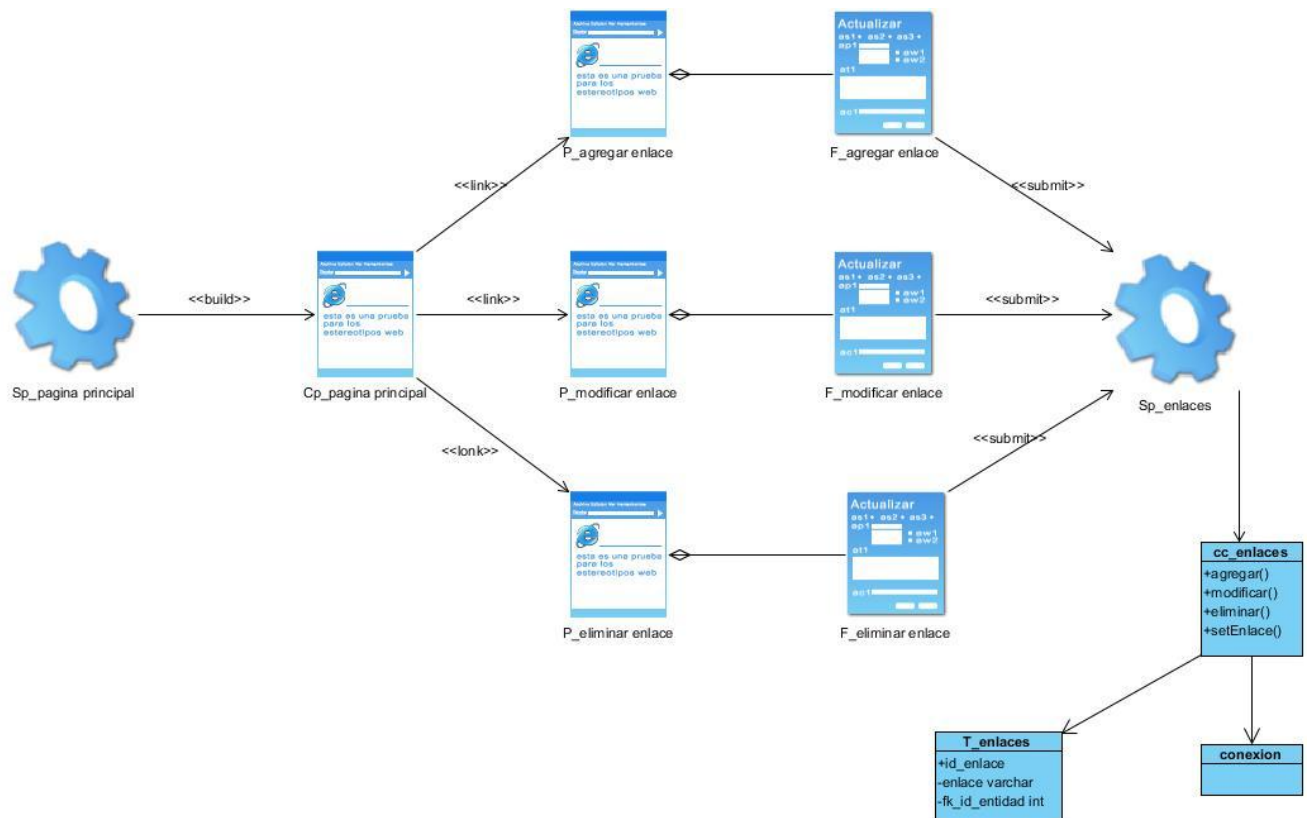


Figura 22. Diagramas de clases de diseño. Caso de uso Gestionar enlaces.

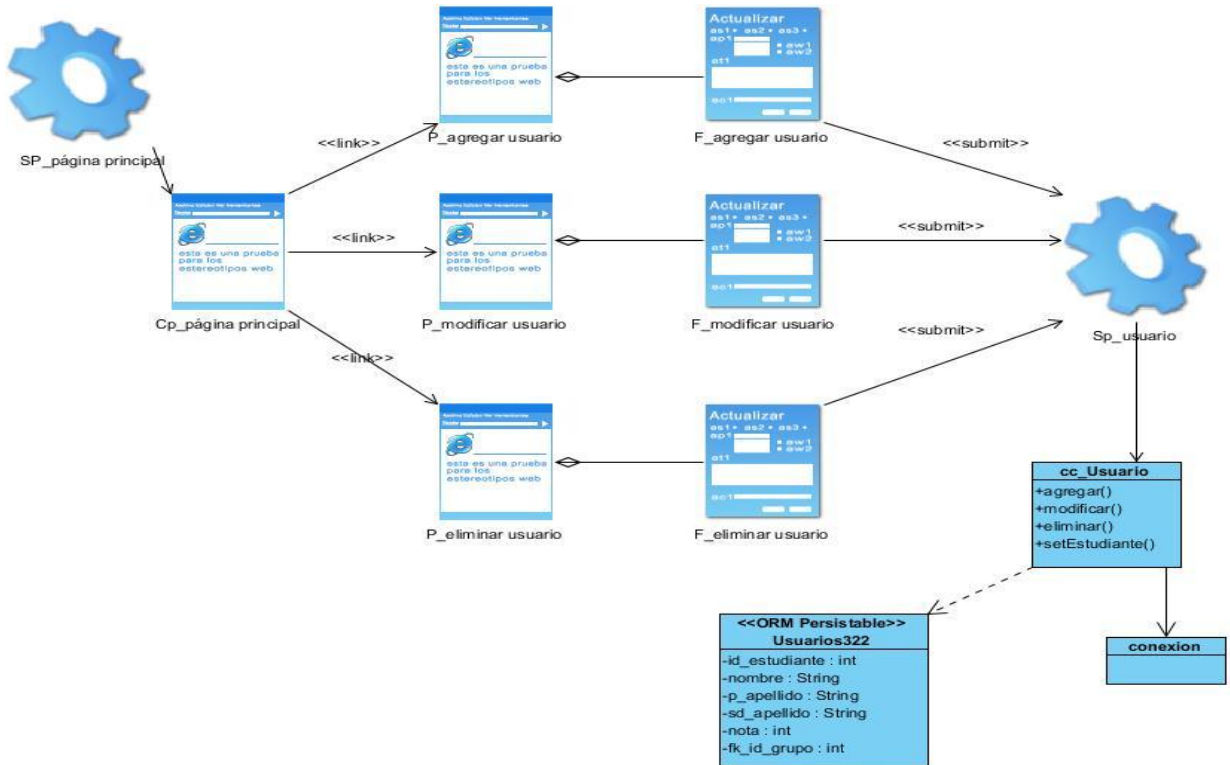


Figura 23. Diagramas de clases de diseño. Caso de uso Gestionar Usuarios.

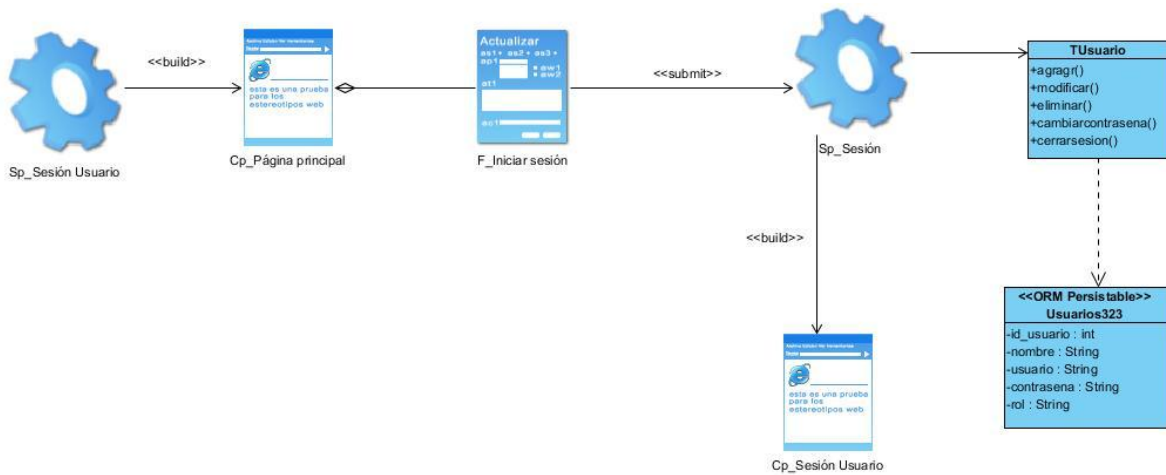


Figura 24. Diagramas de clases de diseño. Caso de uso Iniciar Sesión.

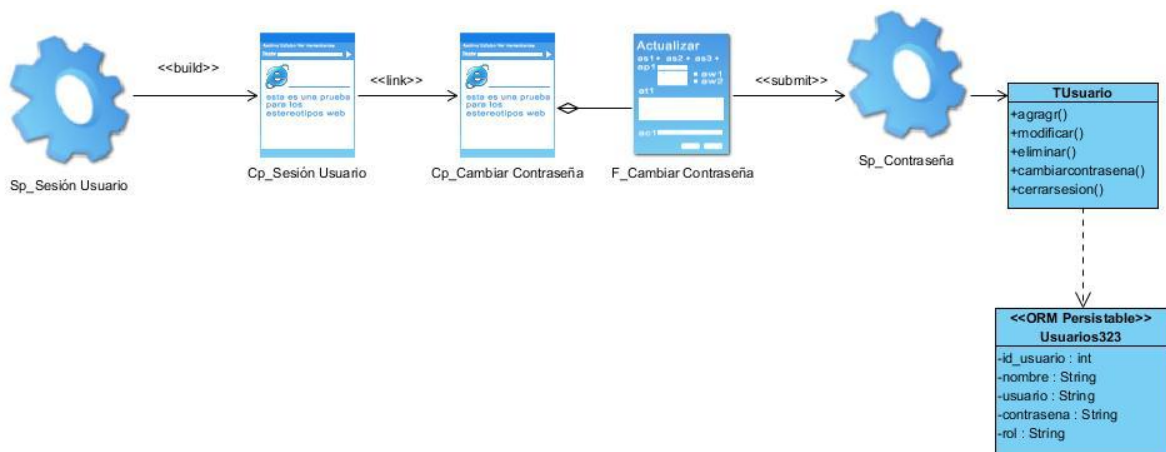


Figura 25. Diagramas de clases de diseño. Caso de uso Cambiar contraseña.

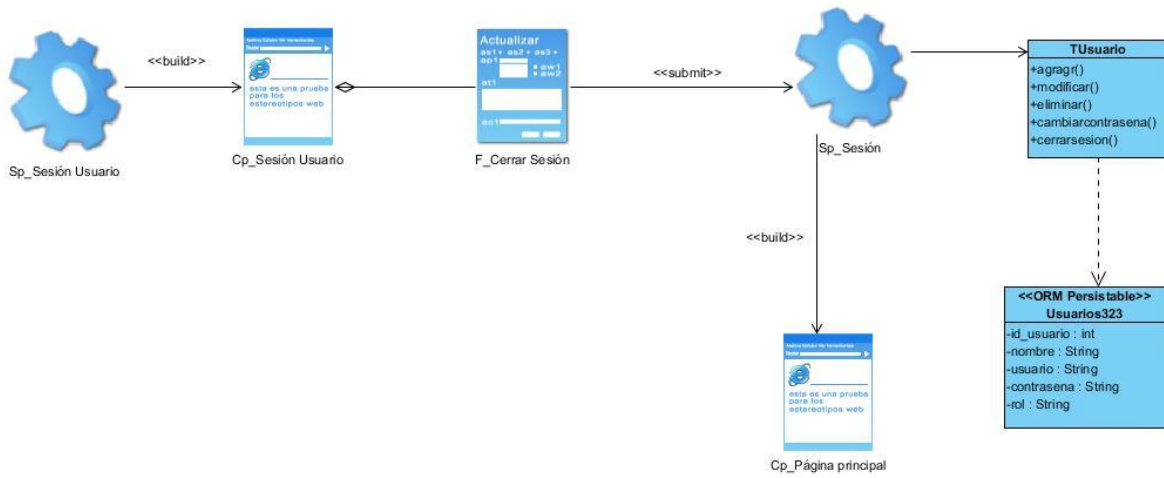


Figura 25. Diagramas de clases de diseño. Caso de uso Cerrar Sesión.

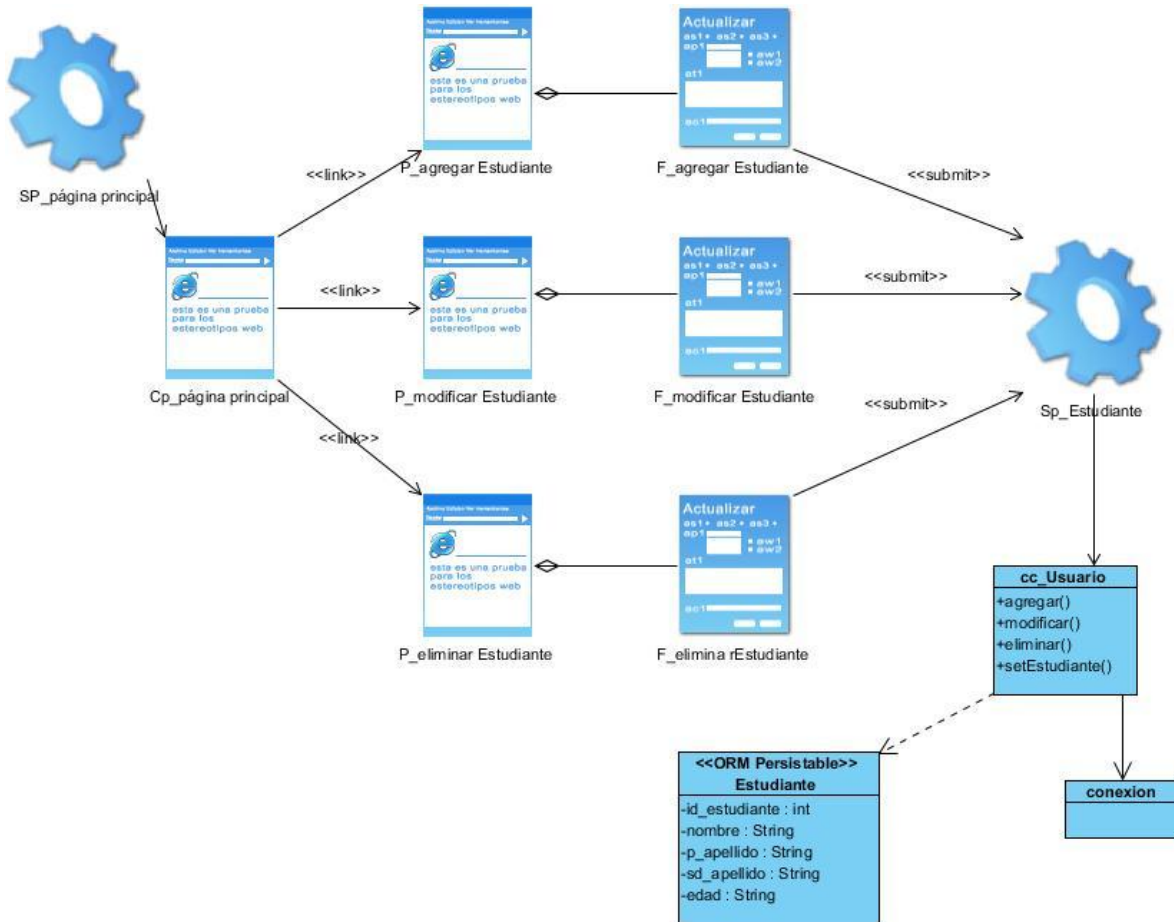


Figura 26. Diagramas de clases de diseño. Caso de uso Gestionar estudiante.

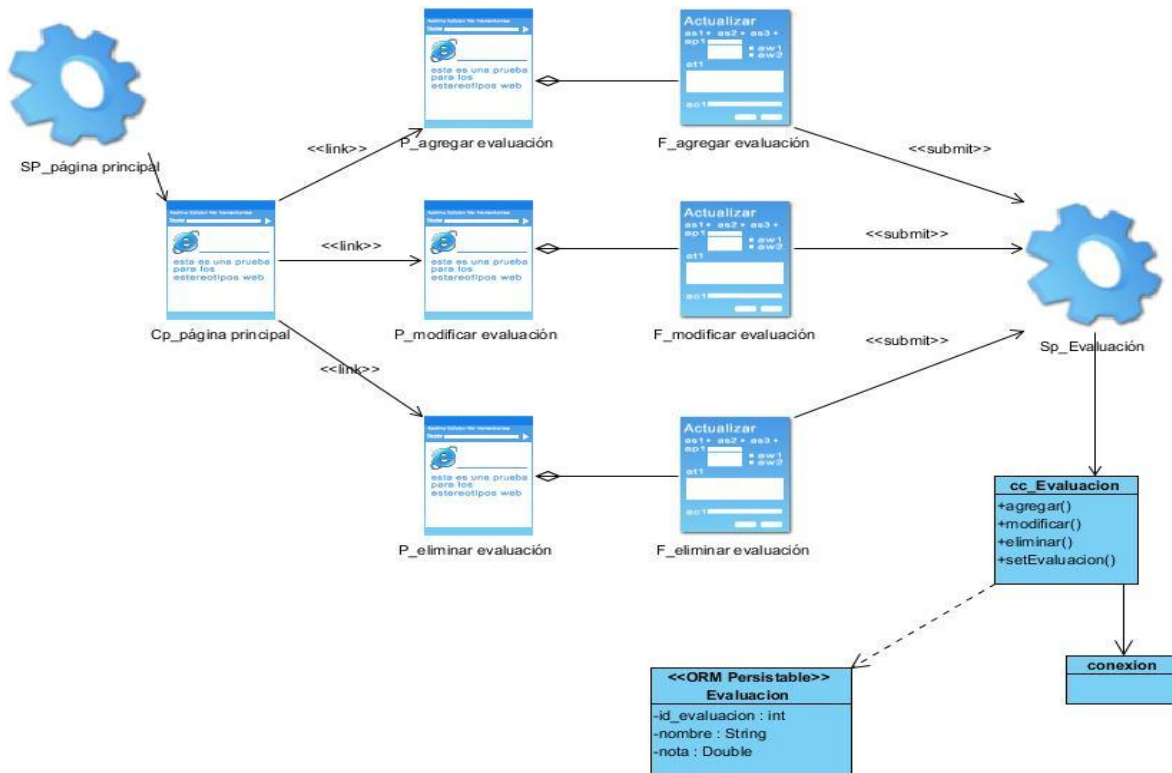


Figura 27. Diagramas de clases de diseño. Caso de uso Gestionar Evaluación.

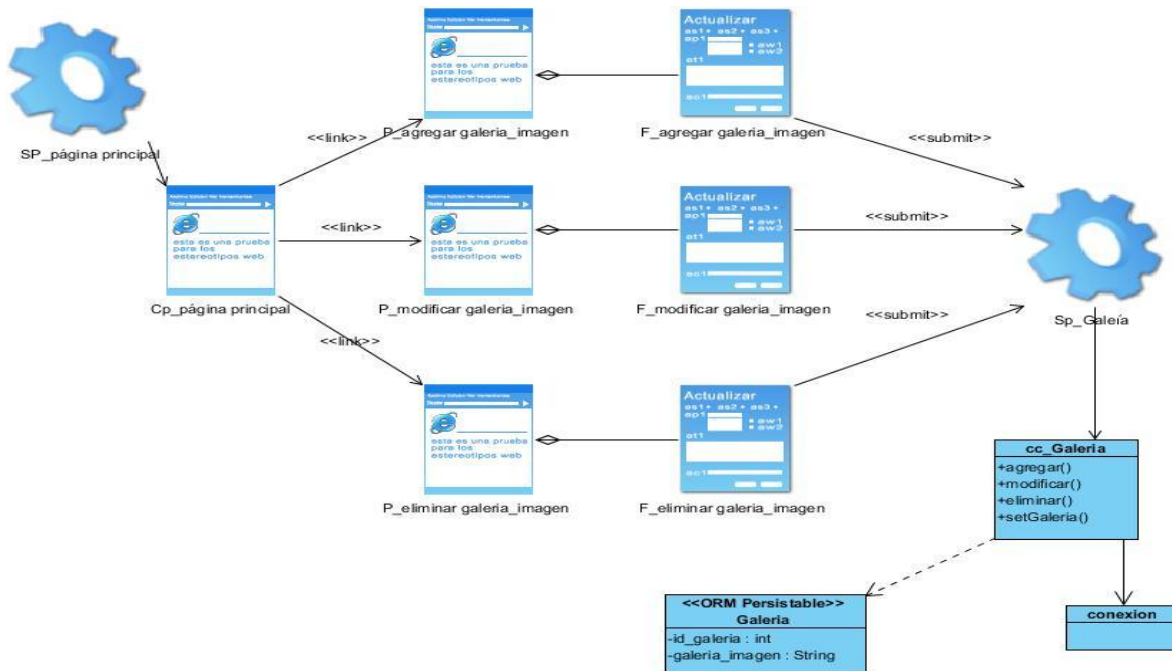


Figura 27. Diagramas de clases de diseño. Caso de uso Gestionar Galería Imagen.

3.2 Diseño de la base de datos

Cuando se va a realizar el diseño de la base de datos para un sistema dado, es necesario determinar los datos que se deben tomar en cuenta y las dependencias funcionales existentes entre ellos. Esto se obtiene luego de realizada la etapa de análisis del sistema y partiendo de lo obtenido en ésta (García, 2006).

3.2.1 Diagrama de clases persistentes

El diagrama del modelo lógico de datos o diagrama de clases persistentes, muestra las clases capaces de mantener su valor en el espacio y en el tiempo (Jacobson, Booch, & Rumbaugh, 2006).

A partir de este planteamiento se definieron las clases que participan en el modelo lógico de datos.

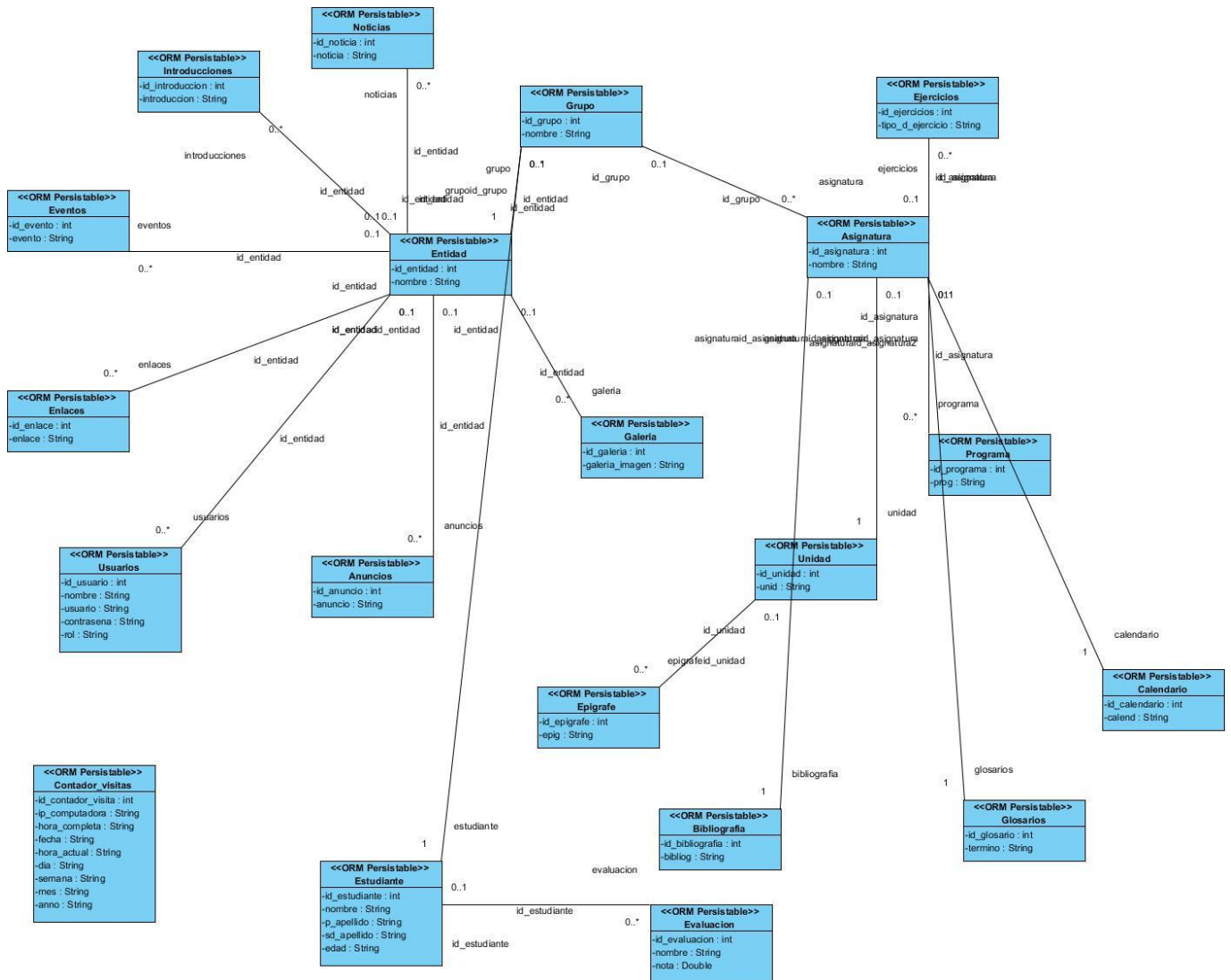


Figura 23. Diagrama de Clases persistentes.

3.2.2 Diagrama del Modelo Físico de Datos

El modelo físico de datos o modelo de datos, representa la estructura o descripción física de las tablas de la base de datos y se obtiene a partir del modelo lógico de datos (Jacobson, Booch, & Rumbaugh, 2006).

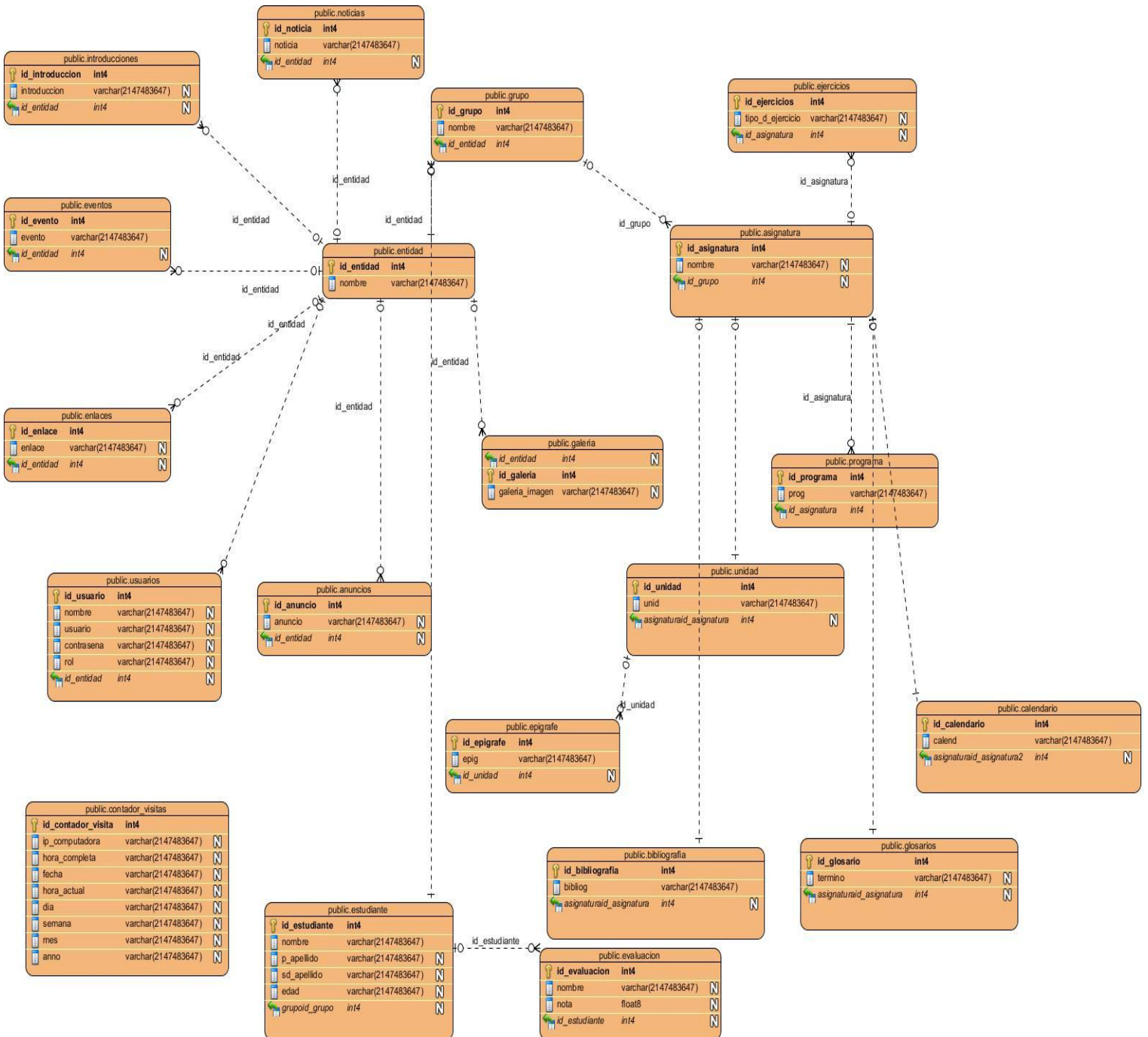


Figura 24. Diagrama de modelo Físico de Datos

3.3 Principios de diseño

El diseño de la interfaz de un Sitio Web, la concepción de la ayuda y el tratamiento de excepciones tiene gran influencia en el éxito o fracaso del mismo. A continuación se describen los principios de diseño seguidos para el desarrollo del sistema en cuestión.

3.3.1 Estándares en la interfaz de la aplicación

Cuando un usuario visita por primera vez un Sitio Web, la impresión que dicho usuario se lleva está dada por el atractivo de su diseño. Es por ello que, para lograr una apariencia adecuada y que el usuario se sienta cómodo, se tienen en cuenta varios aspectos, sobre todo relacionados con tipografía, colores, y navegación.

La interfaz diseñada para el sistema presenta las siguientes características:

- Se utilizarán las tonalidades de azul por ser suaves y refrescantes.
- El vocabulario no utiliza términos técnicos, acercándose al utilizado por los usuarios.
- Información legible.
- La navegación y el uso del sistema es fácil.
- La entrada de información por parte de los usuarios se realiza a través de formularios.
- Las interacciones con el sistema se basan fundamentalmente en selecciones de menú, en sus distintas formas y en acciones sobre elementos de código visual como botones, y mensajes.
- Los procedimientos que se ejecutan para acceder a la información almacenada en la base de datos son rápidos e incrementales obteniéndose resultados inmediatos.

3.3.2 Concepción general de la ayuda

Dentro del mundo de los Sitios Web en general, la ayuda constituye una parte importante del sistema. Las tendencias actuales apuntan a que estas no deben ser muy detallistas o extensas, sino simplemente explicaciones sencillas y aclaraciones del producto y de las operaciones que puede realizar el usuario sobre el mismo. Se concibió una ayuda amigable y práctica para el módulo del sistema relacionado con la entrada y actualización de los datos. La ayuda de la aplicación quedó formada por un menú interno que contiene las funcionalidades del sistema. El usuario podrá navegar por cada uno de ellos y consultar la funcionalidad que realiza, descrita de un modo fácil de entender.

3.3.3 Tratamiento de errores

En el sistema propuesto se evitan, minimizan y tratan los posibles errores, con el fin de garantizar la integridad y confiabilidad de la información que se registra y muestra en el sistema. Las posibilidades de introducir información errónea en el sistema por parte del usuario son mínimas, para lograr esto se realizaron dos tipos de validaciones la del cliente y la del servidor. Las validaciones del lado del cliente se realizaron en Javascript, evitando hacer peticiones innecesarias al servidor, en el caso de que los script no puedan ser validados en el cliente, entonces la validación se realiza en el servidor. En cualquiera de

los dos casos los mensajes de error que emite el sistema se muestran en un lenguaje claro, preciso y de fácil comprensión para los usuarios.

3.3.4 Seguridad

Para lograr la seguridad y protección de la información almacenada, el sistema exige a los usuarios un nombre de usuario y una contraseña para poder acceder al módulo de entrada y actualización de los datos, controlándose de esta forma los niveles de acceso a la información. Solo podrán actualizar información presente en la base de datos del sistema, los usuarios que cuenten con dichos privilegios. La consistencia de los datos es otro aspecto que se toma en consideración, y para ello el sistema cuenta con formularios validados, con funciones del lenguaje PHP que garantizan que la información que se registre en la base de datos sea totalmente consistente e íntegra.

3.4 Modelo de implementación

El modelo de implementación denota la implementación del sistema en términos de componentes y subsistemas de implementación. Describe cómo se organizan los componentes de acuerdo con los mecanismos de estructuración, y modularización disponibles en el entorno de la implementación y en el lenguaje o lenguajes de programación utilizados, y cómo dependen los componentes unos de otros (Cáceres, 2005).

3.4.1 Diagrama de despliegue

El diagrama de despliegue muestra la forma en que los componentes se desplegarán en el sistema, en este se detalla las especificaciones de la red, del servidor y los requisitos de hardware. Según Jacobson, Booch, & Rumbaugh un diagrama de despliegue es un diagrama que muestra la configuración de los nodos que participan en la ejecución y de los componentes que residen en ellos (Jacobson, Booch, & Rumbaugh, 2006).

A continuación se muestra el Diagrama de Despliegue de la aplicación:

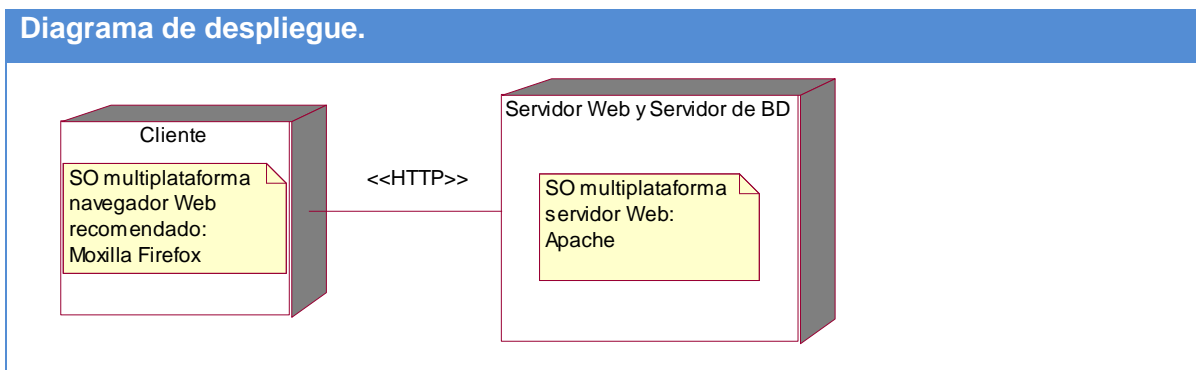


Figura 25. Diagramas de despliegue.

3.4.2 Diagrama de componentes

Un diagrama de componentes muestra la organización y las dependencias entre un conjunto de componentes. Los diagramas de componentes cubren la vista de implementación estática de un sistema. Se relacionan con los diagramas de clases en que un componente se corresponde, por lo común, con una o más clases, interfaces o colaboraciones (Jacobson, Booch, & Rumbaugh, 2006).

La siguiente figura muestra el diagrama de componentes:

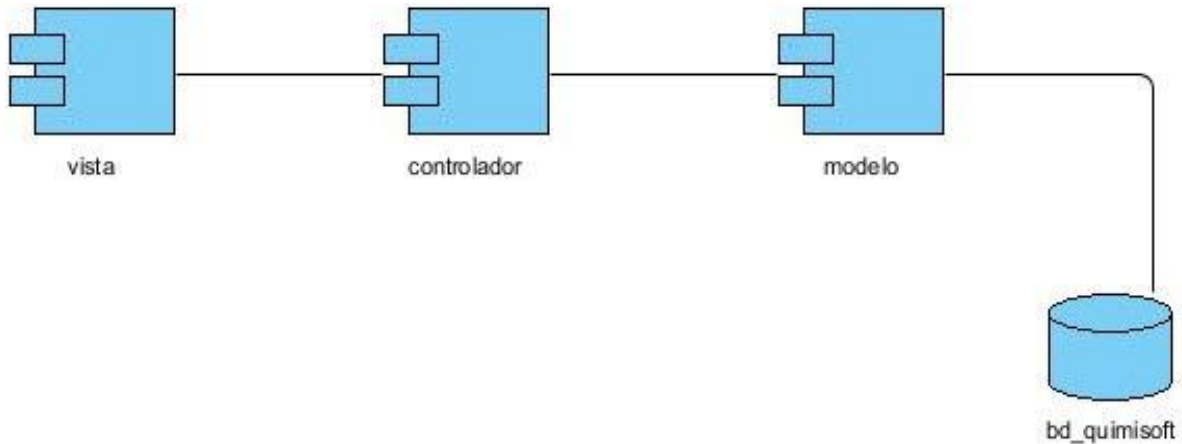


Figura 26. Diagrama de Componentes.

Conclusiones parciales del capítulo

En el capítulo en cuestión se pueden apreciar el resultado de la etapa de implementación del sistema. Se desarrollaron los diagramas de clases, el diseño de la base de datos, el diagrama de despliegue y el de componentes. Además, se describieron los principios de diseño seguidos, específicamente, estándares de la interfaz de usuario, el formato de reportes, el tratamiento de excepciones, la concepción de la ayuda y los estándares de codificación.

Conclusiones:

Los fundamentos teóricos y metodológicos sustentaron la elaboración de una Aplicación Web para contribuir al proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura del oncono grado en la enseñanza preuniversitaria en Cuba. Se adoptaron las metodologías, las herramientas y los lenguajes adecuados para el diseño e implementación de dicho sitio.

Se diseñó la Aplicación Web para contribuir al proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura del oncono grado en la enseñanza preuniversitaria en Cuba para los profesores y estudiantes. Se adoptó como metodología de desarrollo al RUP, con el UML para el modelado asistido por diagramas, utilizando para su modelado la herramienta Visual Paradigm. Se seleccionaron los actores del sistema y del negocio, y sus respectivos casos de usos.

Se implementó una Aplicación Web para apoyar el proceso de enseñanza - aprendizaje de la Ciencia Química como asignatura del oncono grado en la enseñanza preuniversitaria en Cuba, utilizando el lenguaje PHP para la lógica del negocio y los lenguajes Java Script y HTML para la interfaz del usuario. Se utilizó para ello la herramienta Dreamweaver 8, a PostgreSQL como gestor de base de datos por ser de código abierto, pues este ofrece las ventajas de mantenimiento, reusabilidad, escalabilidad y robustez que demande el sistema en el futuro. De manera lógica, el código de la implementación se organizó en tres partes interrelacionadas mediante el patrón de diseño conocido como MVC

Recomendaciones

1. Poner a prueba el sistema durante un período de tiempo significativo para comprobar su desempeño.
2. Ampliar el marco de la aplicación y no dejarlo solamente en el proceso de enseñanza-aprendizaje de la ciencia química como asignatura del onceno grado, sino que ampliar su uso para otras asignaturas y grados diferentes.
3. Ampliar su funcionalidad con nuevos módulos de contenido.

Bibliografía

Álvarez, M. A. (2005). *Evaluando Zend Studio*. Obtenido de Evaluando Zend Studio:
<http://www.maestrosdelweb.com/editorial/zendstudio/>

Álvarez, R. B., & Mayo, I. C. (2009). Las tecnologías de la información y la comunicación en la educación superior. *Revista Iberoamericana de Educación* .

Arquitectura Cliente/Servidor. (2007). Obtenido de Arquitectura Cliente/Servidor:
www.desarrolloweb.com/.../arquitectura-cliente-servidor.html

Cruz Rodríguez, Y., y col. (2010). Herramienta didáctica de las TIC dirigida a la preparación docente metodológica de los estudiantes en la universidad cubana. *Gestiópolis*.

Borges, T. R. (2007). *LA EVOLUCIÓN DE LAS TELECOMUNICACIONES EN SANCTI SPÍRITUS*. SANCTI SPÍRITUS.

Campos, S. A. (2009). *Evolución de la enseñanza asistida por computadoras*.

Castellano, D. (2002). *Aprender y enseñar en la escuela: una concepción desarrolladora*. Ciudad de la Habana.

Castellanos, A. G. (2005). *Comparación entre sistemas de gestión de bases de datos (SGBD)*. Bogotá.

Castillo, F. S. (2008). *La investigación sobre medios de enseñanza: Pasado y Presente*.

Catoní, M. L. (25 de enero de 2009). *Software y Educación*. Obtenido de Software y Educación:
<http://blogs.rimed.cu/infoedu/category/software-educativo/>

Conociendo algo de software educativo. (2009). Obtenido de Conociendo algo de software educativo:
<http://www.enlaces.udec.cl>

Eugenia, M. (26 de junio de 2005). *Las TICS en la Educación*. Recuperado el 2013, de Las TICS en la Educación: <http://educatics.blogspot.com/>

García, R. M. (2006). *Sistema de Base de Datos*. La Habana: Félix Varela.

Hernández. (1989). Recuperado el 05 de febrero de 2013, de
<http://www.infor.uva.es/~descuder/docencia/pd/Node24.html>

Historia de la Informática en América Latina y el Caribe: Investigaciones y testimonios. (2010). Obtenido de Historia de la Informática en América Latina y el Caribe: Investigaciones y testimonios:
<http://dc.exa.unrc.edu.ar/historia/node/31>

Jacobson, I., Booch, G., & Rumbaugh, J. (2006). *El Proceso Unificado de Desarrollo de Software*. Felix varela.

Lago, R. (abril de 2007). *Patrones de diseño software*. Obtenido de Patrones de diseño software: <http://www.proactiva-calidad.com/java/patrones/mvc.html>

Laguna, M. A. (2004). *Ingeniería del Software II*. Recuperado el 12 de abril de 2013, de Ingeniería del Software II: <http://www.infor.uva.es/mlaguna/is2/1-Introd.pdf>

Marquez, P. (2010). *El software educativo*. Barcelona: Universidad Autónoma de Barcelona.

Martínez, I. G. (2009). *Medios de transmisión de las información*.

Pressman, R. S. (2007). *Ingeniería del Software Un enfoque práctico*. La Habana: Félix Varela.

Romero, S. B. (2004). *El Mapa Conceptual como estrategia de enseñanza y aprendizaje en la resolución de problemas*. La Habana.

Rosario, J. (2005). *La Tecnología de la Información y la Comunicación (TIC). Su uso como Herramienta para el Fortalecimiento y el Desarrollo de la Educación Virtual*. Recuperado el 29 de enero de 2013, de La Tecnología de la Información y la Comunicación (TIC). Su uso como Herramienta para el Fortalecimiento y el Desarrollo de la Educación Virtual: <http://www.cibersociedad.net/archivo/articulo.php?art=218>

Anexos

The screenshot shows the 'Agregar Usuario' form in a web application. At the top, there is a navigation bar with 'Inicio', 'Calendario', 'Grupo', and 'Galería de Imágenes'. Below it, a secondary bar contains 'Administración', 'Dirección', 'Gestión', and 'Reportes'. The user is logged in as 'Administrador' and has links for 'Cambiar contraseña', 'Perfil', and 'Salir'. On the left, a sidebar lists menu items: 'Asignatura', 'Programa', 'Unidad', 'Epígrafe', 'Bibliografía', 'Glosario de Términos', and 'Ejercicios'. A 'Noticias' section features a photo of a classroom and the text 'El mundo reconoce'. The main content area is titled 'Usuarios' and includes search and action buttons: 'Buscar', 'Agregar', 'Editar', and 'Borrar'. The 'Agregar Usuario' form contains fields for 'Nombre', 'Usuario', 'Contraseña', 'Confirmar contraseña', 'Nivel de acceso' (a dropdown menu), and 'Entidad' (another dropdown menu). 'Aceptar' and 'Cancelar' buttons are at the bottom of the form. On the right, an 'Eventos' section displays 'Curso de Química Provincial' on '12/05/2014' with an image of chemistry glassware. Below that, an 'Enlaces de interés' section lists 'Correo UNISS' and 'Ignacio Agramonte y Loinaz'.

Figura 27.CU Agregar Usuario.

The screenshot shows the 'Cambiar Contraseña' form. The navigation and sidebar elements are identical to the previous screenshot. The main content area is titled 'Cambiar Contraseña' and features a key icon. The form includes a 'Usuario' dropdown menu set to 'Administrador', and three password fields: 'Contraseña', 'Nueva contraseña', and 'Confirmar contraseña', all masked with dots. 'Aceptar' and 'Cancelar' buttons are at the bottom. The right-hand side of the page, including the 'Eventos' and 'Enlaces de interés' sections, remains the same as in the previous screenshot.

Figura 28.CU Cambiar Contraseña.

Inicio | Calendario | Grupo | Galería de Imágenes

Asignatura
Programa
Unidad
Epígrafe
Bibliografía
Glosario de Términos
Ejercicios

Noticias




«Páginas»
1 2 [Sig.](#)

El mundo reconoce los resultados de la salud pública

Eventos

Cuncurso de Química Provincial
12/05/2014



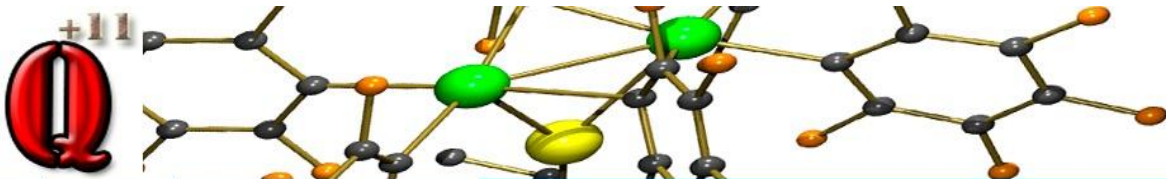
Enlaces de interés

[Correo UNISS](#)
[Ignacio Agramonte y Loínez](#)

Login

Usuario
ernestorl@uniss.ed
Contraseña
●●●●●●●●
[Login](#)

Figura 29 CU Iniciar Sesión.




Inicio | Calendario | Grupo | Galería de Imágenes

Administración | Dirección | Gestión | Reportes

Bienvenido, Administrador | [Cambiar contraseña](#) | [Perfil](#) | [Salir](#)


Asignatura
Programa
Unidad
Epígrafe
Bibliografía
Glosario de Términos
Ejercicios

Noticias



Eventos

Buscar | Agregar | Editar | Borrar


X Nombre	Descripción
Cuncurso de Química Provincial 12/05/2014	

Se ha mostrado 1 coincidencia de 1

«Páginas»
1

Eventos

Cuncurso de Química Provincial
12/05/2014



Enlaces de interés

[Correo](#)

Figura 30. Ventana Principal.



Inicio | Calendario | Grupo | Galería de Imágenes

Administración | Dirección | Gestión | Reportes

Bienvenido, Administrador | [Cambiar contraseña](#) | [Perfil](#) | [Salir](#)

Asignatura

Programa

Unidad

Epigrafe

Bibliografía

Glosario de Términos

Ejercicios

Noticias



El mundo reconoce

Asignatura

Buscar | Agregar | Editar | Borrar

<input checked="" type="checkbox"/> Asignatura	Grupo
<input type="checkbox"/> Química	11no1

Se ha mostrado 1 coincidencia de 1 «Páginas»

1

Eventos

Curso de Química Provincial
12/05/2014



Enlaces de interés

[Correo UNISS](#)

[Ignacio Agramonte y Loínez](#)

Figura 31. CU Gestionar Asignatura.



Inicio | Calendario | Grupo | Galería de Imágenes

Administración | Dirección | Gestión | Reportes

Bienvenido, Administrador | [Cambiar contraseña](#) | [Perfil](#) | [Salir](#)

Asignatura

Programa

Unidad

Epigrafe

Bibliografía

Glosario de Términos

Ejercicios

Noticias



El mundo reconoce

Bibliografía

Buscar | Agregar | Editar | Borrar

<input checked="" type="checkbox"/> Bibliografía	Asignatura
<input type="checkbox"/> Semiconductores	Química
<input type="checkbox"/> Polímeros	Química

Se ha mostrado 2 coincidencias de 3 «Páginas»

1

Eventos

Curso de Química Provincial
12/05/2014



Enlaces de interés

[Correo UNISS](#)

[Ignacio Agramonte y Loínez](#)

Figura 32. CU gestionar Bibliografía.



Inicio | Calendario | Grupo | Galería de Imágenes

Administración | Dirección | Gestión | Reportes

Bienvenido, Administrador | [Cambiar contraseña](#) | [Perfil](#) | [Salir](#)

Asignatura
Programa
Unidad
Epígrafe
Bibliografía
Glosario de Términos
Ejercicios

Ejercicios

Buscar | Agregar | Editar | Borrar

<input checked="" type="checkbox"/> Ejercicios	Asignatura
<input type="checkbox"/> Laboratorio de química	Química
<input type="checkbox"/> Ejercicios de concurso	Química

Se ha mostrado 2 coincidencias de 2
«Páginas»
1


Noticias



El mundo reconoce

Eventos

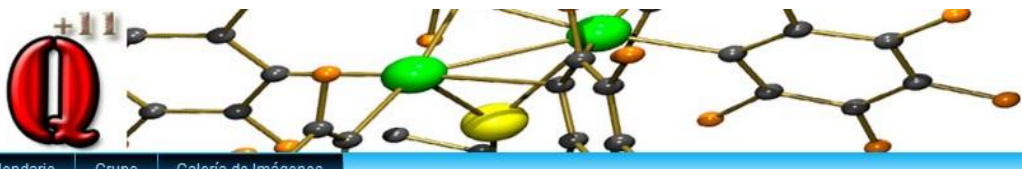
Concurso de Química Provincial
12/05/2014



Enlaces de interés

[Correo UNISS](#)
[Ignacio Agramonte y Loínaz](#)

Figura 33. CU Gestionar Ejercicios.



Inicio | Calendario | Grupo | Galería de Imágenes

Administración | Dirección | Gestión | Reportes

Bienvenido, Administrador | [Cambiar contraseña](#) | [Perfil](#) | [Salir](#)

Asignatura
Programa
Unidad
Epígrafe
Bibliografía
Glosario de Términos
Ejercicios


Estudiante

Buscar | Agregar | Editar | Borrar

<input checked="" type="checkbox"/> Nombre	Primer Apellido	Segundo Apellido	Grupo
Se ha mostrado 0 coincidencia de 0			

No se ha encontrado ninguna coincidencia en la base de datos.


Noticias



El mundo reconoce

Eventos

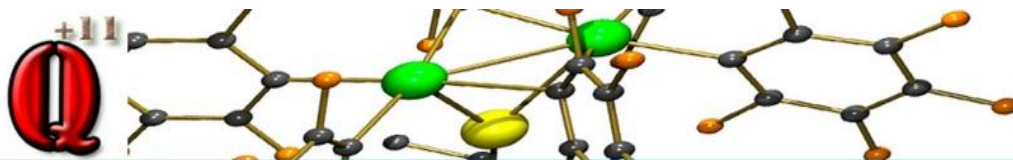
Concurso de Química Provincial
12/05/2014



Enlaces de interés

[Correo UNISS](#)
[Ignacio Agramonte y Loínaz](#)

Figura 34. CU Gestionar Estudiante.



Inicio | Calendario | Grupo | Galería de Imágenes

Administración | Dirección | Gestión | Reportes

Bienvenido, Administrador | [Cambiar contraseña](#) | [Perfil](#) | [Salir](#)

Asignatura
Programa
Unidad
Epígrafe
Bibliografía
Glosario de Términos
Ejercicios


Calendario

X Calendario Asignatura

SEMANA	Nº de clase	CONTENIDO	ACT.EXP.	EVAL.
2-8 Sept.	1	Introducción al curso.		
	2	UNIDAD 1. Sistemas dispersos. Disoluciones	Demost.	
	3	Proceso de disolución de las sustancias.		
9-15 Sept.	4	Cambios energéticos en el proceso de disolución.	Demost.	ECS
	5	ejercitación	PL.	
	6	Factores relacionados con la velocidad de disolución. De los sólidos en los líquidos		
16-22		P. Laboratorio. Factores		

16. 22


Noticias



El mundo reconoce

Eventos

Concurso de Química Provincial
12/05/2014



Enlaces de interés

[Correo](#)

Figura 35. CU Gestionar Calendario.



Inicio | Calendario | Grupo | Galería de Imágenes

Administración | Dirección | Gestión | Reportes

Bienvenido, Administrador | [Cambiar contraseña](#) | [Perfil](#) | [Salir](#)

Asignatura
Programa
Unidad
Epígrafe
Bibliografía
Glosario de Términos
Ejercicios

Anuncios

X Nombre Descripción

Escuela avierta próximo lunes




Se ha mostrado 1 coincidencia de 1 «Páginas»

Noticias



El mundo reconoce

Eventos

Concurso de Química Provincial
12/05/2014



Enlaces de interés

[Correo UNISS](#)

[Ignacio Agramonte y Loínez](#)

Figura 36. CU Gestionar Anuncios.



Inicio Calendario Grupo **Galería de Imágenes**

Administración Dirección Gestión Reportes

Bienvenido, Administrador [Cambiar contraseña](#) [Perfil](#) [Salir](#)

Asignatura
Programa
Unidad
Epígrafe
Bibliografía
Glosario de Términos
Ejercicios

Eventos

	Nombre	Descripción
<input checked="" type="checkbox"/>	Concurso de Química Provincial 12/05/2014 	Se ha mostrado 1 coincidencia de 1 «Páginas» 1

Eventos

Concurso de Química Provincial
12/05/2014



Enlaces de interés

[Correo UNISS](#)

[Ignacio Agramonte y Loínaz](#)

Noticias



El mundo
reconoce

Figura 37. Gestionar Eventos.



Inicio Calendario Grupo **Galería de Imágenes**

Administración Dirección **Gestión** Reportes

Bienvenido, Administrador [Cambiar contraseña](#) [Perfil](#) [Salir](#)

Asignatura
Programa
Unidad
Epígrafe
Bibliografía
Glosario de Términos
Ejercicios

Eventos

	Nombre	Descripción
<input checked="" type="checkbox"/>	Inicio	Es una necesidad de la sociedad cubana la incorporación de nuevos conceptos a partir del perfeccionamiento del sistema socialista cubano y de la insuficiente cultura económica que hasta el momento hemos desarrollado, por lo que es de vital importancia educar a las nuevas generaciones en una cultura económica. Por lo antes expuesto debemos establecer una estrategia que permita abordar los diferentes ejes temáticos de la Cultura económica, utilizando las diferentes vías que en la práctica escolar se presentan, sin excluir la preparación que requieren los directivos y docentes para asumir la labor educativa en este tema. Se ha mostrado 1 coincidencia de 1 «Páginas» 1

Eventos

Concurso de Química Provincial
12/05/2014



Enlaces de interés

[Correo UNISS](#)

[Ignacio Agramonte y Loínaz](#)

Noticias



El mundo
reconoce

Figura 38. Devolver reportes (pdf) de: -Datos Estudiante – Evaluación Estudiante.