



FACULTAD DE INGENIERIA INFORMÁTICA

TESIS EN OPCIÓN A LA CATEGORÍA DOCENTE DE MÁSTER EN TECNOLOGÍAS DE LA
INFORMACIÓN Y LAS COMUNICACIONES EN LA EDUCACIÓN

TÍTULO: "PLATAFORMA PARA EL CONTROL DEL USO DE SOFTWARES EDUCATIVOS"

AUTOR: ING. KADIR HECTOR ORTIZ

TUTOR: MSC. DIONEL A. ACOSTA DUARTE

AÑO 49 DE LA REVOLUCIÓN

2007

Índice

Introducción.....	1
I Capítulo I: “Fundamentación Teórica”	5
I.1 Proceso de enseñanza aprendizaje.....	5
I.2 Enseñanza asistida por computadora (EAC).....	8
I.3 Formas de aplicación de la computadora en la enseñanza.....	12
I.4 Integración de las TIC en la Educación	13
I.5 Tecnología Multimedia.....	14
I.5.1 Funciones pedagógicas de los sistemas multimedia	14
I.5.2 La aplicación de la multimedia en la EAC.....	17
I.6 Plataformas existentes en los entornos educativos.....	20
I.7 Nuevas tecnologías de acceso a la información.....	21
I.7.1 Servicios Web	21
I.7.2 WSDL.....	22
I.7.3 XML.....	23
I.7.4 SOAP	25
Conclusiones.....	25
II Capítulo II: “Análisis y Diseño de la Solución”	28
II.1 Descripción del Modelo de Dominio	28
II.1.1 Definición de los objetos y los conceptos principales.....	28
II.1.2 Representación del Modelo de Objetos de Dominio.....	29
II.1.3 Reglas del negocio a considerar.....	29
II.2 Descripción del Sistema Propuesto.....	30
II.2.1 Concepción general del sistema.....	30
II.2.2 Requerimientos Funcionales.....	30
II.2.3 Requerimientos no Funcionales.....	31
II.3 Modelo de Casos de Uso del Sistema.....	33
II.3.1 Actores del sistema.....	33
II.3.2 Casos de Uso del Sistema.....	34
II.3.3 Descripción de los Casos de Uso del Sistema.....	36

II.4	Factibilidad Económica	45
II.4.1	Planificación por puntos de función.....	47
II.4.2	Determinación de los costos	52
II.4.3	Beneficios tangibles e intangibles	57
II.4.4	Análisis de costos y beneficios	57
II.4.5	Tratamiento de Excepciones.....	57
II.5	Modelo de Clases Web.....	57
II.6	Diseño de la Base de datos.....	58
II.6.1	Diagrama del modelo lógico de datos.....	59
II.6.2	Diagrama del modelo físico de datos.....	59
II.7	Diagrama de Implementación.....	59
	Conclusiones	60
III	Capítulo III: “Análisis de los Resultados”	61
III.1	Descripción General del Software	61
III.1.1	Validación	61
III.1.2	Cuestionario.....	62
III.1.3	Muestreo Aleatorio Simple	63
	Conclusiones	68
IV	Conclusiones.....	69
V	Recomendaciones	70
VI	Referencias Bibliográficas	71
VII	Bibliografías	75

Índice de figuras

Fig 1 Diagrama de Objetos del Dominio	29
Fig 2 Diagramas de Casos de Uso del Sistema.....	35
Fig 3 Diagrama de Implementación	60
Fig 4 Cuestionario. Pregunta 1	64
Fig 5 Cuestionario. Pregunta 2	65
Fig 6 Cuestionario. Pregunta 3	66
Fig 7 Cuestionario. Pregunta 4	67

Índice de Tablas

Tabla 1 Actores del Sistema	34
Tabla 2 Descripción del caso de uso de sistema: Autenticar Estudiante	36
Tabla 3 Descripción del caso de uso de sistema: Autenticar Trabajador	37
Tabla 4 Descripción del caso de uso de sistema: Insertar datos de usuario	37
Tabla 5 Descripción del caso de uso de sistema: Actualizar datos de usuario	38
Tabla 6 Descripción del caso de uso de sistema: Cambiar estado usuario.....	39
Tabla 7 Descripción del caso de uso de sistema: Recuperar datos de usuario.....	39
Tabla 8 Descripción del caso de uso de sistema: Cambio de contraseña.....	40
Tabla 9 Descripción del caso de uso de sistema: Asignar permisos a usuario.....	40
Tabla 10 Descripción del caso de uso de sistema: Insertar datos de software educativo.....	41
Tabla 11 Descripción del caso de uso de sistema: Mostrar softwares educativos	41
Tabla 12 Descripción del caso de uso de sistema: Actualizar datos de software educativo.....	42
Tabla 13 Descripción del caso de uso de sistema: Cambiar estado SE.....	42
Tabla 14 Descripción del caso de uso de sistema: Insertar datos de visitas de estudiantes a software	43
Tabla 15 Descripción del caso de uso de sistema: Insertar datos de visitas a contenido de software.	43
Tabla 16 Descripción del caso de uso de sistema: Asignar asignatura a	44
Tabla 17 Descripción del caso de uso de sistema: Actualizar asignatura a profesor .	44
Tabla 18 Descripción del caso de uso de sistema: Mostrar Asignatura.....	45
Tabla 19 Planificación: Entradas Externas	48
Tabla 20 Planificación: Peticiones	48
Tabla 21 Planificación: Ficheros Internos.....	51
Tabla 22 Planificación: Puntos de Función	52
Tabla 23 Planificación: Miles de Instrucciones Fuertes	52
Tabla 24 Costos: Factores de escalas.....	53
Tabla 25 Costos: Factores de Ajuste.....	55
Tabla 26 Diagramas de Clases Web.	58

RESUMEN

El presente trabajo constituye un esfuerzo más por desarrollar una interacción educando-educador acorde a las necesidades de los tiempos modernos. Es reconocida la importancia que tiene para cualquier proceso docente educativo que el profesor conozca a fondo toda la información referente a dicho proceso.

Como resultado de la presente investigación se desarrolló una plataforma basada en la Arquitectura de Servicios que utiliza el nuevo paradigma de los servicios web para lograr una aplicación independiente de plataforma. El objetivo de dicha plataforma es servir de puente entre el comportamiento de los estudiantes en el uso de los Softwares Educativos, desarrollados para complementar el proceso de aprendizaje, y los profesores que son los encargados de evaluar dicho comportamiento.

Esta experiencia, en sus inicios, está siendo llevada a cabo en la Facultad de Informática de la Universidad de Cienfuegos. Esta investigación lleva como nombre: "Plataforma de Control de Uso de Software Educativo" (PCUSE).

Introducción

Es de vital importancia para cualquier proceso docente educativo que el profesor conozca a fondo toda la información referente a dicho proceso. Datos como “conocimiento de base que posee un estudiante”, “cantidad de conocimiento nuevo por unidad de tiempo que se puede adquirir”, “capacidad de análisis de los estudiantes”, “nivel de aceptación e interés sobre determinados temas”, entre otros, permiten desarrollar una interacción educando-educador acorde a las necesidades de los tiempos modernos.

En nuestro país, por una parte, se han realizado grandes esfuerzos en el desarrollo de softwares educativos que acercan al estudiante de una manera más amena al conocimiento necesario en su proceso de aprendizaje. Por otra parte, se han implementado entornos para controlar el uso de la información disponible en las redes informáticas. Ambos esfuerzos contribuyen a mejorar el proceso docente educativo pero no son capaces de explotar las nuevas tecnologías de la información en toda su dimensión. En el primer caso se pierde, muchas veces, la interacción que se puede lograr entre: profesor-software educativo-estudiante. En el segundo caso se refiere a entornos desarrollados para una plataforma específica que imposibilita que una gama amplia de estos softwares educativos (SE), entiéndase multimedias, hipermedias y otros tipos, puedan ayudar al profesor a controlar, supervisar y estudiar las diferentes variables que afectan y/o influyen en la interacción mencionada anteriormente.

Problema

Las plataformas existentes en la esfera educativa no permiten mantener un control estricto y centralizado sobre el acceso de los estudiantes a los softwares educativos independientemente del entorno en el que estos hayan sido creados.

Hipótesis

La creación de una plataforma basada en las nuevas tecnologías de acceso a las redes informáticas permitirá mantener un control estricto y centralizado del acceso de los estudiantes a los softwares educativos, independientemente del sistema operativo y/o lenguaje de programación y contribuirá a perfeccionar el proceso enseñanza-aprendizaje.

Variable Dependiente

Plataforma para el control del acceso de los estudiantes a los softwares educativos.

Variables Independientes

- 1 Unidad, tema, epígrafe, subepígrafe y tópico consultado.
- 2 Respuesta ante las preguntas (si existen)
- 3 Usuario

Objetivo General

Desarrollar un entorno basado en las nuevas tecnologías de acceso a la información, que siendo independiente de la plataforma permita registrar datos sobre el uso y consumo de la información de los softwares educativos.

Pregunta de la Investigación

¿Se puede obtener y almacenar datos de un software educativo a través de un entorno multiplataforma e independiente del lenguaje, a través de las redes de comunicación?

Tareas de la Investigación

- 1 Estudiar las nuevas tecnologías de la información y las comunicaciones.
- 2 Estudiar las tecnologías de desarrollo de aplicaciones y programación.
- 3 Estudiar y analizar la literatura psicológica, pedagógica y metodológica vinculada con el problema a tratar.
- 4 Investigar las características de los Softwares Educativos.

- 5 Analizar las plataformas educativas orientadas a la enseñanza.
- 6 Diseñar y elaborar un entorno que permita obtener y almacenar datos a partir de un software educativo.

Con el objetivo de desarrollar las tareas planteadas se utilizaron los siguientes métodos de investigación:

Método Teórico:

- 1 Estudio y análisis de la literatura pedagógica y metodología vinculada con el problema a tratar.
- 2 Estudio de los documentos normativos y metodológicos del Ministerio de Educación, así como otros materiales relacionados con el tema.

Método de Entrevista:

Se realizaron entrevistas a profesores y estudiantes:

- 1 A los profesores para conocer el sistema de evaluación.
- 2 A los estudiantes para conocer como hacen uso de softwares educativos.

Método de Escritura de Ideas

Se recolectaron ideas a partir de las entrevistas realizada a los profesores y estudiantes.

Método de Análisis y Síntesis

Para establecer nexos, comparar resultados, determinar enfoques comunes y aspectos distintivos de los diferentes enfoques estudiados, lo que permite arribar a conclusiones.

Aporte Práctico

La creación del entorno que permitirá el registro de los datos sobre el uso y la interacción de los estudiantes con los softwares educativos

.

Novedad Científica

La posibilidad de almacenar datos sobre la interacción de los estudiantes con los softwares educativos, los cuales le servirán al profesor para mejorar el proceso de enseñanza aprendizaje.

El presente documento está estructurado en 3 capítulos

Capítulo 1: “Fundamentación Teórica”: En este capítulo se aborda la fundamentación teórica del tema y los conceptos asociados al dominio del problema. Se explica en detalles el problema a resolver.

Capítulo 2: “Análisis y Diseño de la Solución”. En este capítulo se presenta el análisis y el diseño de la solución propuesta utilizando la metodología RUP (Rational Unified Process por sus siglas en inglés: Proceso Unificado de Desarrollo), se hace un análisis de los costos y beneficios con un estudio de factibilidad económica.

Capítulo 3: “Análisis de los Resultados”. En este capítulo se presenta la validación de los resultados obtenidos a partir de la aplicación de una encuesta a desarrolladores de softwares educativos.

Capítulo I: "Fundamentación Teórica"

En este capítulo se hace un estudio del estado del arte, analizando bibliografías sobre el desarrollo del proceso de enseñanza aprendizaje, sobre el uso de las computadoras en el proceso docente educativo, enfatizando en el impacto de las nuevas tecnologías en los estudiantes. Así como estudio y selección de las tecnologías, metodologías y lenguajes de programación a utilizar.

I.1 Proceso de enseñanza aprendizaje

Como proceso de enseñanza - aprendizaje se define "el movimiento de la actividad cognoscitiva de los alumnos bajo la dirección del maestro, hacia el dominio de los conocimientos, las habilidades, los hábitos y la formación de una concepción científica del mundo" [1]. Se considera que en este proceso existe una relación dialéctica entre profesor y estudiante, los cuales se diferencian por sus funciones; el profesor debe estimular, dirigir y controlar el aprendizaje de manera tal que el alumno sea participante activo, consciente en dicho proceso, o sea, "enseñar" y la actividad del alumno es "aprender".

Algunos autores [2][3][4] consideran como componentes del proceso de enseñanza a los objetivos, el contenido, los métodos, los medios y su organización los que conforman una relación lógica interna.

Los medios de enseñanza son considerados el sostén material de los métodos y están determinados, en primer lugar, por el objetivo y el contenido de la educación, los que se convierten en criterios decisivos para su selección y empleo.

La relación maestro - alumno ocupa un lugar fundamental en este contexto del proceso docente - educativo; el maestro tiene una función importante y los medios de enseñanza multiplican las posibilidades de ejercer una acción más eficaz sobre los alumnos.

Varios son los investigadores que se han dedicado a profundizar en el estudio de la

teoría sobre los medios de enseñanza, su definición, metodología de aplicación, influencias en el proceso de asimilación de conocimientos y desarrollo de habilidades, entre otros temas.

La definición de medios de enseñanza ha sido abordada por diferentes autores; por ejemplo, Lothar Klingberg los concibe como "...todos los medios materiales necesarios para el maestro o el alumno, para una estructuración y conducción efectiva y racional del proceso de educación e instrucción...".

El colectivo de autores [5] define "...los medios de enseñanza constituyen distintas imágenes y representaciones de objetos y fenómenos que se confeccionan especialmente para la docencia...", este concepto enmarca los medios de percepción directa, imágenes fijas y audiovisuales, excluyendo otros como los medios de laboratorio, equipos sonoros, las computadoras, entre otros.

El autor Vicente González Castro [6] define: "...Los medios de enseñanza son todos aquellos componentes del proceso docente - educativo que le sirven de soporte material a los métodos de enseñanza para posibilitar el logro de los objetivos planteados". En esta definición se reconoce como medios de enseñanza tanto a los medios visuales y sonoros como a los objetos reales, a los libros de textos, los laboratorios y a todos los recursos materiales que sirven de sustento al trabajo del maestro.

Dentro de los múltiples medios de enseñanza que se emplean en la actualidad, resulta de especial interés para este trabajo la computadora, principalmente su uso e influencia sobre el proceso de enseñanza - aprendizaje.

En las tesis de García de la Vega [7] y Pérez Fernández V. [8], que expresa "...Las computadoras son máquinas de enseñar universales, que permiten realizar a un nivel cualitativamente superior las funciones de todos los medios técnicos creados hasta el momento con fines docentes..."; este concepto enmarca las potencialidades del

medio dentro del proceso de enseñanza, aspecto que se debe tener en cuenta a la hora de su utilización en la docencia.

En el análisis comparativo de otros medios técnicos de enseñanza, como los audiovisuales o material impreso con la computadora, ésta los aventaja por su capacidad de interactuar con el estudiante, unido a la posibilidad de uso de sonido, imágenes y videos al mismo tiempo, así como la capacidad de almacenamiento de altos volúmenes de información y su velocidad de ejecución; estas características la convierte en un medio de alta potencialidad en la docencia.

El creciente empleo de la computadora en la enseñanza ha ido en aumento en el mundo. Willem J. Pelgrum, señala "...la cantidad media de máquinas disponibles en los establecimientos docentes aumentó de manera gradual en los años 80, aunque en algunos países hubo bruscos aumentos, resultantes de programas oficiales de estímulo...", esto evidencia la creciente incorporación de la computadora en la enseñanza y Cuba no ha sido una excepción en este proceso.

Sin embargo la eficiencia del uso de la computadora en la docencia no depende solo del número de equipos instalados sino también del tipo de software que se utiliza. Alessi [9], Garay [10] García [11], Pérez [12] y O'Shea [13], plantean que los Entrenadores, Tutoriales, Juegos didácticos, Sistemas expertos para la enseñanza, Simuladores de experimentos y los Evaluadores, son las formas más difundidas de presentarse el software con fines docentes.

Los programas bien diseñados pueden fomentar el aprendizaje conceptual, al ofrecer un contexto que permita a los alumnos extraer los principios generales de los contenidos. Este tipo de enseñanza, "descubrimiento guiado", puede ser muy útil para el aprendizaje de ciertas materias.

I.2 Enseñanza asistida por computadora (EAC)

La combinación de tres factores: profesor, alumno y computadora han abierto nuevos caminos, métodos de trabajo e ideas de acción en la pedagogía, la Enseñanza Asistida por Computadora (EAC). En ella se destaca el uso de la computadora como elemento importante en el perfeccionamiento de la enseñanza.

El creciente desarrollo de la técnica informática en la actualidad, ha permitido elevar el nivel de exigencia de los softwares educativos.

En la actividad docente, los softwares se pueden manifestar a partir de los objetivos para lo que son diseñados. Alessi [14] los clasifica como: Tutoriales, Entrenadores, Simuladores, Juegos instructivos, Test u otros. Entre ellos se hará especial énfasis en los Tutoriales y los Entrenadores, por estar especialmente relacionados con el presente trabajo.

Los orígenes de la educación basada en computadoras se remontan a los años 20, cuando el Dr. Sidney L. Pressey de la Universidad de Ohio, desarrolló su máquina de enseñar, la primera referencia sobre esta máquina de enseñar fue hecha por el Dr. Pressey en un artículo publicado en *School and Society* en 1926. La generalización de las máquinas de enseñar tuvo su período álgido después de la Segunda Guerra Mundial en gran parte por los trabajos de B.F. Skinner, de la Universidad de Harvard, a mediados de la década de 1950.

Las máquinas de enseñar por su complejidad iban desde una caja de cartón hasta grandes y costosas máquinas electrónicas, el modesto progreso de estas máquinas fue descrito por [15].

Las máquinas de enseñar y la enseñanza programada representaron métodos educativos integrados por un conjunto de principios didácticos de valor psicológico. Estos principios y consideraciones tienen un valor incuestionable. El desarrollo de la Enseñanza Programada fue encabezado por:

B.F.Skinner: psicólogo norteamericano, inició la teoría que sirvió de base a la enseñanza programada, sobre la base de grandes trabajos experimentales adiestrando animales. La esencia de su teoría es el principio de compensación según el cual la concesión de un incentivo, ante una respuesta correcta, incrementa la fuerza de la enseñanza. El modelo de programa elaborado por Skinner es lineal por lo que no contempla la posibilidad de cambios en la secuencia de ejecución en correspondencia con las individualidades del alumno.

Norman A. Crowder: psicólogo norteamericano que propuso una forma alternativa de la enseñanza programada que consiste en una estrategia ramificada en la que cada respuesta del alumno conduce por un determinado camino, por lo que Crowder la denominó "programación intrínseca". Este modelo es cualitativamente superior al de Skinner. Crowder limitaba las preguntas al tipo de "elección múltiple".

La utilización de las computadoras en proyectos educativos comenzó en los años 60. Evidentemente se apoyaron en los resultados alcanzados en la enseñanza programada. En una primera etapa se implementaron programas lineales según el modelo de Skinner.

La capacidad de las computadoras de tomar decisiones lógicas permitió introducir rápidamente la programación intrínseca. Aunque la realización de programas ramificados con ayuda de los lenguajes de programación tales como BASIC, PASCAL, C no entraña gran dificultad, se pensó ello superaba la competencia, o cuando menos los intereses de la mayoría de los autores de material de enseñanza para computadoras y esto provocó el desarrollo de una rama especial de los lenguajes de programación denominada lenguajes de autor para crear la ilusión de que los autores podían escribir programas de enseñanza sin hacer realmente programación de computadoras.

Posteriormente según O'Shea [16] se desarrollaron:

Programas generativos: Se basan en que en ciertas situaciones los alumnos aprenden mejor enfrentándose a problemas de dificultad apropiada que atendiendo a explicaciones sistemáticas. El método exige que el sistema genere problemas, soluciones y diagnósticos asociados. Este enfoque se reveló apropiado solo para algunos casos.

Modelos matemáticos de aprendizaje: Se basan en el uso de teorías estadísticas sobre el aprendizaje. Son de aplicabilidad limitada pues únicamente en algunos casos especiales se pueden expresar los procesos de aprendizaje en la forma requerida.

Proyecto TICCIT: Iniciado en 1972 y destinado a la educación de adultos, el modelo TICCIT (Time Shared Interactive Computer Instructional Television) se basa en dos premisas fundamentales. La primera es que la enseñanza se realiza de forma expositiva, explicándose directamente al alumno los diferentes hechos, reglas y conceptos. Dicha exposición es seguida por ejemplos y prácticas. La segunda premisa es que el secuenciamiento de la enseñanza es controlado por el alumno, tanto en relación a la materia en su conjunto como a cualquier parte de la misma. El alumno controla el orden de presentación de las explicaciones, ejemplos, ejercicios prácticos, comprobaciones e incluso el nivel de dificultad del material. Este modelo aparece descrito en detalle en [17].

Proyecto PLATO: Estaba basado en el sistema PLATO (Programmed Logic for Automatic Teaching Operation) de la Universidad de Illinois, este consistía en redes muy extendidas de terminales y a diferencia del TICCIT no era realizada por equipos de producción sino que se suministraba un lenguaje de autor TUTOR para que los profesores lo usaran, por lo que la calidad del material de enseñanza era muy variable.

Simulación: La simulación es un proceso que genera un ambiente representativo del objeto de estudio, que ayuda al alumno a construir una imagen que modela el objeto

de referencia mediante la experimentación.

Juegos: Los juegos instructivos son programas con un alto poder de motivación, debido a que por lo general ponen en marcha mecanismos para estimular la fantasía y plantear retos. Dichos mecanismos garantizan la ejercitación que desarrolla y consolida un conjunto de habilidades. Los mejores éxitos de esta modalidad se han obtenido en el trabajo con niños.

Resolución de problemas: se deriva de la teoría de Piaget y de la Inteligencia Artificial (IA). Se basa en la hipótesis de que el esfuerzo mental requerido para escribir un programa de computadora ayuda al desarrollo de la técnica de resolución de problemas de índole general. Se utiliza la computadora como un ambiente de aprendizaje (mundos).

Modalidades liberadoras: Los alumnos pueden usar la computadora para resolver ellos mismos los problemas sin tener que escribir ellos mismos los programas, es decir usándola como un instrumento de trabajo que ahorra tiempo y trabajo en la realización de cálculos, gráficos, recuperación de información, procesamiento de texto.

Sistemas de diálogo (generativos): Se basan en la utilización de estrategias tutoriales y de una interfaz de lenguaje natural. Son sistemas de iniciativa propia que utilizan representaciones bastante complejas del conocimiento.

Todos estos modelos evidentemente son de valor, pero hasta el surgimiento de la microcomputadora a finales de los 70 el uso de las computadoras estaba solo al alcance de un grupo élite de universidades, ya que su instalación completa era muy cara, y su utilización quedaba justificada en muy pocos casos además de la investigación.

Con el desarrollo sostenido del hardware, el aumento de su potencia tanto en la velocidad de procesamiento, como en la capacidad de almacenamiento de la información, la aparición de la tecnología multimedia, han hecho factible la

implementación de sistemas que antes no pasaban de ser un sueño con pálidas implementaciones.

En la actualidad, el entusiasmo con las nuevas posibilidades tecnológicas ha hecho que en muchas ocasiones se ignoren las consideraciones didácticas y pedagógicas, tal como era en las edades negras de la EAC cuando los aspectos técnicos eran considerados más importantes que la calidad instructiva del material. [18].

Es evidente que el surgimiento de las microcomputadoras, el aumento sostenido de su potencia aparejado al decrecimiento casi constante de los precios han hecho posible que muchos países puedan ir extendiendo la aplicación de la computación en la enseñanza.

I.3 Formas de aplicación de la computadora en la enseñanza.

Actualmente existen varias formas de aplicación de la computadora en la enseñanza, se pueden considerar como clásicas las siguientes:

1 Tutorial:

Es la forma más conocida, se caracteriza por el establecimiento de un diálogo que garantiza que el alumno pueda ir relacionando lo ya conocido con lo nuevo, para mejorar la eficiencia del proceso enseñanza - aprendizaje [19]. El programa tutor ofrece información, pregunta para determinar el grado de comprensión y toma decisiones en función de las respuestas. Debido a que el objetivo de este trabajo es la confección de un tutorial.

Ejercitación reproductiva:

La ejercitación reproductiva es una forma utilizada para consolidar el nivel reproductivo de la asimilación. Se parte de un conjunto de preguntas relacionadas con el objeto de estudio y se conforma una lista para establecer el orden de presentación que el profesor considera más conveniente. En la medida que las preguntas son respondidas

correctamente, se borran de la lista. En caso contrario se altera la lista, reincorporando las preguntas mal respondidas en posiciones preseleccionadas. El proceso se repite hasta obtener una lista vacía. La ejercitación reproductiva se basa en la teoría sustentada por algunos investigadores [20], que plantea que el hombre opera con dos formas de memoria. Una de corta permanencia, y otra de permanencia prolongada. La información puede almacenarse con facilidad en la primera, pero se pierde también con facilidad. La repetición facilita la transferencia de información de la memoria de corta permanencia a la de permanencia prolongada.

- Simulación: Ver tópico anterior.
- Juegos: Ver tópico anterior.
- Entrenamiento: Los programas entrenadores tienen como función ejercitar en proporción a los objetivos de la etapa de aprendizaje por la que se transita ante la mirada virtual del profesor. De esta forma se pretende combinar la participación activa del estudiante con la acción correctora del profesor. De los resultados de la ejercitación se puede obtener un diagnóstico sobre el estado de los conocimientos de los estudiantes, para facilitar información que permita adaptar la estrategia pedagógica del profesor a las características individuales de los estudiantes.
- Búsqueda de información: El aprendizaje está indisolublemente ligado a la búsqueda de información, y la computadora puede facilitarla en gran medida usando un ambiente diseñado para ello. Para esto se utiliza lo que se conoce como hipertexto. Un hipertexto bien diseñado puede posibilitar el acceso rápido y fácil a los conceptos fundamentales.

I.4 Integración de las TIC en la Educación

Los profesionales de la educación pueden aprovechar las nuevas posibilidades que proporcionan las TIC para impulsar este cambio hacia un nuevo paradigma educativo más personalizado y centrado en la actividad de los estudiantes. Además de la necesaria alfabetización digital de los alumnos y del aprovechamiento de éstas para la mejora de la productividad en general, la cual constituye poderosas razones para aprovechar las posibilidades de innovación metodológica que ofrecen las TIC para

lograr una escuela más eficaz e inclusiva. [21]

I.5 Tecnología Multimedia

Los dispositivos para navegar en la Web han evolucionado en los últimos años de forma exponencial. Junto a esta evolución, ha quedado atrás la introducción simple de texto e imágenes, siendo sustituida por la posibilidad de incorporar una gran variedad de objetos, como pueden ser archivos de sonido, vídeo y animación, con el objetivo de enriquecer la experiencia del usuario y ofrecer al mismo tiempo la posibilidad de utilizar diferentes medios conjuntamente.

Pero la introducción de todos estos elementos de forma compatible se convierte en una labor complicada para desarrolladores. Por este motivo, las especificaciones desarrolladas por el W3C trabajan en la creación de lenguajes en los que esta multimedia enriquecida sea compatible con los diversos dispositivos y navegadores existentes en el mercado.

Por lo tanto, las Tecnologías Multimedia son el producto de una demanda creciente en lo que a incorporación conjunta y compatibilidad de dispositivos y objetos se refiere.

I.5.1 Funciones pedagógicas de los sistemas multimedia

Autores como Fernández, B (1987), González, V (1986) y Klingberg, L (1978) han hecho aportes a las funciones de los medios de enseñanza en el proceso pedagógico. Partiendo que las funciones se evidencian en el funcionamiento externo de un objeto el sistema multimedia responde a las siguientes funciones: cognoscitiva, comunicativa, motivadora, informativa, integrativa, sistematizadora, y de control

En la función cognoscitiva tomamos como punto partida el criterio expresado por Klingberg, L (1978) cuando señala que estructurar el proceso de aprendizaje como un proceso del conocimiento requiere el empleo de medios de enseñanza, y por supuesto el sistema multimedia es uno de ellos. Este sistema actúa cumpliendo con el principio del carácter audiovisual de la enseñanza, y de esta manera permite

establecer el camino entre las representaciones de la realidad objetiva en forma de medios y los conocimientos que asimilarán los estudiantes.

La función comunicativa, está apoyada en el papel que los medios de enseñanza cumplen en el proceso de la comunicación. En el mismo, ocupan el lugar del canal que es a su vez soporte de la información, es vínculo portador del mensaje (González V. 1989) que se trasmite a los estudiantes. Por tal razón el sistema multimedia actúa como soporte a partir del cual se desarrolla el proceso comunicativo, entre los realizadores del mismo y los estudiantes que lo emplean. Es en ese momento donde la multimedia manifiesta la interactividad con el estudiante. El puede seleccionar la información, el camino, la multimedia le puede sugerir otras vías y otras fuentes alternativas o no a las que pretende tomar. La interacción es parte de la función comunicativa pues con ella se logra la verdadera comunicación con el sistema. Este proceso no debe verse solamente entre el sistema y los estudiantes sino que se extiende a las posibilidades de comunicación telemática con otros profesores, estudiantes, o centros remotos, situación que no es cumplida por otros medios hasta el presente.

En la función informativa partimos del punto de vista de Fernández, B [22] al expresar de que el empleo de los medios permite brindar una información más amplia, completa y exacta, ampliando los límites de la transmisión de los conocimientos. La aplicación del sistema multimedia enriquece el proceso de transmisión de la información que es necesario en la educación, debido a la integración de medios, a las posibilidades de búsquedas de información fuera del propio sistema, a las consultas con el profesor y otros alumnos, así como a la interactividad entre el sistema y el estudiante.

La función integradora es una de las más importantes de este medio, pues la misma se refleja en otras de las funciones que ya hemos explicado. La integración de medios no significa la sustitución de ellos, ni la sobrevaloración de este medio por encima de otros. Pero en la enseñanza es importante facilitar al estudiante el acceso

a la información, el ahorro de tiempo y la disminución del esfuerzo en el aprendizaje. Estas necesidades las cumple el sistema multimedia al permitir la integración de numerosos medios; de esta manera el estudiante no tiene que buscar en el libro la tarea, en el casete de audio escuchar la grabación o ver la animación en el video, pues todos ellos estarán integrados en el propio sistema. Pero esta función además se extiende a la integración de los contenidos.

La función sistematizadora obedece a la planificación del trabajo con la multimedia, la que se cumple desde la etapa de elaboración del mismo. Aún cuando este medio se caracteriza por la navegación no lineal, ello no significa en modo alguno que el aprendizaje sea improvisado. La sistematización garantiza que el estudiante pueda ir ampliando sus conocimientos a medida que avanza en el trabajo con el multimedia y a su vez va comprobando lo aprendido.

La multimedia manifiesta su función de control a partir de la posibilidad que tiene el estudiante de comprobar su aprendizaje, y el profesor de conocer este. El sistema actúa en la medida que el estudiante avanza y puede colocar preguntas, realizar ejercicios con la finalidad de consolidar y ejercitar. La retroalimentación que el obtiene mediante su auto evaluación le permite además corregir los métodos que emplea, su eficiencia y trazarse nuevas formas de auto enseñanza.

Una misma multimedia ofrece la variante de ser tan útil para el estudiante aventajado como para el que no lo es. El primero podrá ir más rápido, indagar en otras fuentes de información y sentir la necesidad de aprender mas, mientras que el segundo no se sentirá inferior ni marginado, sino que buscará la vía para seguir desarrollándose aunque más lentamente.

Tomando en cuenta lo anterior puede señalarse que los sistemas multimedia rompen con la pasividad en la apropiación de la información que caracterizan a los medios audiovisuales. La observación mecánica que implican los medios audiovisuales desaparece con el empleo de este sistema, al adquirir un carácter heurístico la búsqueda de la información. Con él se pierde lo rutinario, lo mecánico que implica la

observación de los audiovisuales.

El sistema multimedia tiene la capacidad de involucrar al estudiante en su propio aprendizaje y debemos ver el mismo no desde la perspectiva en que hemos observado hasta el presente los distintos medios de enseñanza, sino desde un nuevo ángulo en el que priman tanto la selección del camino a seguir, la interacción con el estudiante como con personas alejadas de nuestro entorno físico. Solo así podemos comprender las enormes posibilidades que tiene este medio y las que se incrementarán en un futuro próximo.

I.5.2 La aplicación de la multimedia en la EAC.

La forma original de presentación de los sistemas de EAC de los 60, que combinaba explicaciones textuales, con sesiones de preguntas y respuestas simples en forma también textual, ha mejorado con el tiempo añadiéndole gráficos, y posteriormente animaciones. Durante los 80 el progreso fue notable, aunque estaba claro que había un largo camino por recorrer hasta lograr el objetivo final de producir Sistemas de EAC que pudieran rivalizar seriamente con los maestros buenos, se sentía que se iba en la dirección correcta [23]

A finales de los 80 surge el concepto de multimedia/hipermedia, aunque existen muchas definiciones acerca de este término se opta por la que plantea que son sistemas que integran en una aplicación al menos tres de los siguientes elementos: texto, gráficos, animaciones, audio y video.

El desarrollo de la multimedia ha aumentado en forma exponencial las posibilidades de transmisión de la información, estas posibilidades han promovido la aplicación de la multimedia en la educación en la creencia de que los programas multimedia pueden ofrecer ambientes de aprendizaje de una riqueza sin paralelos, un ejemplo de este apoyo entusiasta a la multimedia es la anotación siguiente de John Sculley, Director Ejecutivo de Apple Computer Inc:

“Profesores y estudiantes comandarán un ambiente de aprendizaje tan rico, que si me lo hubieran descrito cuando yo estaba en la escuela me habría parecido enteramente mágico. Imaginen un aula con una ventana a todo el conocimiento del mundo. Imaginen un profesor que pueda traer a la vida cualquier imagen, cualquier sonido, cualquier suceso. Imaginen a un estudiante con la posibilidad de visitar cualquier lugar en la Tierra en cualquier momento en la historia. Imaginen una pantalla que puede mostrar en colores vivos, el interior de una célula, el nacimiento y la muerte de estrellas, batallas entre ejércitos, y los triunfos del arte. Y entonces imagine que usted tiene acceso a todo esto, y más, con el solo esfuerzo de pedir que aparezca. Hoy eso parece mágico todavía. Pero la habilidad para proveer este tipo de ambiente de aprendizaje esta todavía dentro de nuestra mente”.

[24]

¿Cuál es la base para una pedagogía efectiva para la multimedia?

Según [25] el candidato líder es la filosofía pedagógica conocida como "constructivismo" y agrega que hay tres principios primarios de la teoría cognitiva contemporánea que son necesarios para liberar el potencial de la multimedia para el aprendizaje:

- Construcción del conocimiento: la multimedia debe diseñarse siguiendo el principio de que el aprendizaje es un proceso de construcción del conocimiento [26] y no de absorción del conocimiento.
- Aprendizaje dependiente del conocimiento: la multimedia debe estructurarse de forma que apoye el principio de que el aprendizaje depende del conocimiento, o sea las personas deben inevitablemente usar el conocimiento que poseen para sobre él construir el nuevo conocimiento.
- Aprendizaje relacionado con la situación: la multimedia debe apoyarse en el principio de que el aprendizaje está fuertemente relacionado con la situación que ocurre, o sea deben introducirse los problemas primero con el contenido y las

habilidades necesarias para la solución y comprensión de estos después, este enfoque le facilitará a maestros y estudiantes enlazar el conocimiento recientemente adquirido en forma de respuestas activas a problemas simulados. [27]

La implementación de los sistemas multimedia se puede hacer usando los lenguajes de programación de propósito general como C++, Basic, Pascal pero el surgimiento de los lenguajes de autor tales como Hypercard (para las Mac), Macromedia Director (Mac y PC), Multimedia ToolBook (PC), Macromedia Flash, etc.. los que han facilitado mucho el desarrollo de sistemas con fines educativos, informativos, etc.

Los sistemas informáticos pueden cumplir con las siguientes funciones: [29]

Función instructiva: está dirigida a la formación en los alumnos, del sistema de conocimientos, capacidades, habilidades y hábitos matemáticos que se corresponden con su etapa de desarrollo. A través de los problemas deben ser fijados conceptos, teoremas y procedimientos matemáticos.

Función desarrolladora: está encaminada a fomentar el pensamiento científico y teórico de los alumnos y a dotarlos de métodos efectivos para la actividad intelectual. Esta es la función rectora en la formación de habilidades por utilizar los métodos del conocimiento científico (la observación, comparación, experimentación, análisis y síntesis, generalización, etc.), como métodos de aprendizaje.

Función educativa: está orientada a la formación de la concepción científica del mundo en los alumnos. También está encaminada al desarrollo de los intereses cognoscitivos y de cualidades de la personalidad, así como a lograr que los alumnos conozcan las realidades del mundo. Posibilita además el desarrollo del patriotismo y el internacionalismo.

Función de control: se orienta a determinar el nivel de cumplimiento de las tres funciones anteriores, o sea, la instrucción y educación de los alumnos, su capacidad para el trabajo independiente, el grado de desarrollo de su pensamiento, es decir a

comprobar en que medida se cumplen los objetivos de la asignatura en el tratamiento de problemas.

En el caso de la función de control nos permite entre otras cosas determinar si un Software Educativo cumple a cabalidad con el objetivo para el cual fue diseñado. En varias ocasiones el SE no provoca el nivel necesario de motivación cognoscitiva, y por tanto, no contribuye al desarrollo intelectual del alumno en toda la dimensión requerida.

I.6 Plataformas existentes en los entornos educativos

En la esfera educativa han surgido muchas herramientas para ayudar al profesor en su trabajo. Dichas herramientas son utilizadas tanto en la educación presencial, como a distancia. Esto se logra a través del uso de las nuevas tecnologías en la educación. Ejemplo de estas plataformas son: CLAROLINE, TELEDUC, ILIAS, GANESHA, FLE3, MOODLE, MICROCAMPUS.

Moodle

Es una plataforma de aprendizaje a distancia basada en software libre. Constituye un sistema de gestión de la enseñanza, es decir, una aplicación diseñada para ayudar a los educadores a crear cursos de calidad en línea. En términos de arquitectura, se trata de una aplicación web que puede funcionar en cualquier computadora, pero no permite retroalimentarse sobre el uso de cualquier tipo de software educativo.

Microcampus

Es un gestor de cursos de última generación para el apoyo de la enseñanza presencial y una herramienta esencial para la enseñanza a distancia a través de Internet. Pertenece a una red de centros en Iberoamérica, que desarrollan proyectos educativos basados en la educación a distancia con el uso de las nuevas tecnologías aplicadas a la formación. Permite conocer en detalles el uso por parte de los estudiantes de la plataforma, pero no contempla el uso de otro software desarrollado

en plataformas diferentes.

CLAROLINE

Es un gestor de educación a distancia que ofrece facilidades de trabajo a estudiantes, profesores y trabajadores de centros educativos, incluye vínculos a sistemas utilizados en el centro donde se usa la plataforma pero no contiene ninguna herramienta que permita conocer el desarrollo de los estudiantes en los softwares educativos.

Se encontraron otras plataformas en el mercado como TELEDUC, ILIAS, GANESHA, FLE3, entre otras, pero ninguna permite salvar información referente al uso y consumo de softwares instalados en la red.

I.7 Nuevas tecnologías de acceso a la información

I.7.1 Servicios Web

Un servicio web es una entidad programable que proporciona alguna funcionalidad determinada, y es accesible a cualquier número de sistemas que usen las normas de Internet. Un servicio web puede ser usado internamente por una aplicación o ser publicado hacia Internet. Estos servicios permiten la ejecución de sus funcionalidades sin importar la plataforma, sistema operativo, o lenguaje en el cual estén implementados. [30]

Los servicios web se pueden utilizar para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios web. Para mejorar la interoperabilidad entre distintas implementaciones de servicios web se ha creado el organismo WS-I, encargado de desarrollar diversos perfiles para definir de manera más exhaustiva estos estándares. [31]

Los servicios web tienen una interfaz descriptiva en un formato que puede ser

procesado por una máquina (específicamente WSDL) y otros sistemas que interactúan con el servicio web utilizando mensajes SOAP.

Los servicios web brindan grandes ventajas dentro de las aplicaciones distribuidas como son:

- 1 Aportan interoperabilidad entre aplicaciones de software independientemente de sus propiedades o de las plataformas sobre las que se instalen.
- 2 Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.
- 3 Al apoyarse en HTTP, los servicios Web pueden aprovecharse de los sistemas de seguridad firewall sin necesidad de cambiar las reglas de filtrado.
- 4 Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados. [32]

Atendiendo a las ventajas, antes mencionadas, que brindan el uso de webservice se incluye su utilización en el desarrollo de este proyecto.

Los servicios web pueden ser utilizados a través de dos protocolos fundamentalmente: RPC y SOAP. El protocolo RPC tiene una dependencia de la plataforma y la tecnología que se utiliza para ejecutar los servicios web, en cambio SOAP es un protocolo que permite la independencia de plataforma y tecnología.

I.7.2 WSDL

WSDL es un formato XML que describe los servicios de red como un conjunto de puntos finales que procesan mensajes contenedores de información orientada tanto a documentos como a procedimientos. Las operaciones y los mensajes se describen de forma abstracta y después se enlazan a un protocolo de red y a un formato de mensaje concreto para definir un punto final de red. Los puntos finales concretos relacionados se combinan en puntos finales abstractos (servicios). WSDL es extensible, lo que permite la descripción de puntos finales de red y sus mensajes, independientemente de los formatos de los mensajes o protocolos de red utilizados

para comunicarse. [33]

I.7.3 XML

El Lenguaje Extensible de Marcas, abreviado XML (eXtensible Markup Language), describe una clase de objetos de datos llamados documentos XML y describe parcialmente el comportamiento de los programas de computadora que los procesan. XML es un "perfil de aplicación" o una forma restringida de SGML, el Lenguaje Estándar Generalizado de Marcación [ISO 8879]. Por construcción, los documentos XML son documentos SGML conformados. [34]

Los documentos XML están compuestos por unidades de almacenamiento llamadas entidades, que contienen tanto datos analizados como no analizados. Los datos analizados están compuestos de caracteres, algunos de los cuales, de la forma datos carácter, y otros de la forma etiquetas. Las etiquetas codifican una descripción de la estructura de almacenamiento del documento y su estructura lógica.

XML fue desarrollado por un Grupo de Trabajo XML (originalmente conocido como "*SGML Editorial Review Board*") formado bajo los auspicios del Consorcio World Wide Web (W3C), en 1996. Fue presidido por Jon Bosak de Sun Microsystems con la participación activa de un Grupo Especial de Interés en XML (previamente conocido como Grupo de Trabajo SGML) también organizado en el W3C. Dan Connolly sirvió como contacto entre el GT y el W3C.

Los objetivos de diseño de XML son [35]:

1. XML debe ser directamente utilizable sobre Internet.
2. XML debe soportar una amplia variedad de aplicaciones.
 - XML debe ser compatible con SGML.
 - Debe ser fácil la escritura de programas que procesen documentos XML.
 - El diseño de XML debe ser formal y conciso.
 - Los documentos XML deben ser fácilmente creables.

XML es la solución a un problema de comunicación entre programas de computadoras. XML intenta ser un formato absolutamente genérico, con el que se puede describir cualquier tipo de archivo.

Características de XML

- 1 Es una arquitectura más abierta y extensible. No se necesitan versiones para que puedan funcionar en futuros navegadores. Los identificadores pueden crearse de manera simple y ser adaptados en el acto en internet/intranet por medio de un validador de documentos (parser).
- 2 Mayor consistencia, homogeneidad y amplitud de los identificadores descriptivos del documento con XML (los RDF Resource Description Framework), en comparación a los atributos de la etiqueta <META> del HTML.
- 3 Integración de los datos de las fuentes más dispares. Se podrá hacer el intercambio de documentos entre las aplicaciones tanto en el propio PC como en una red local o extensa.
- 4 Datos compuestos de múltiples aplicaciones. La extensibilidad y flexibilidad de este lenguaje nos permitirá agrupar una variedad amplia de aplicaciones, desde páginas web hasta bases de datos.
- 5 Gestión y manipulación de los datos desde el propio cliente web.
- 6 Los motores de búsqueda devolverán respuestas más adecuadas y precisas, ya que la codificación del contenido web en XML consigue que la estructura de la información resulte más accesible.
- 7 Se desarrollarán de manera extensible las búsquedas personalizables y subjetivas para robots y agentes inteligentes. También conllevará que los clientes web puedan ser más autónomos para desarrollar tareas que actualmente se ejecutan en el servidor.
- 8 Se permitirá un comportamiento más estable y actualizable de las aplicaciones web, incluyendo enlaces bidireccionales y almacenados de forma externa.
- 9 El concepto de "hipertexto" se desarrollará ampliamente (permitirá denominación independiente de la ubicación, enlaces bidireccionales, enlaces que pueden especificarse y gestionarse desde fuera del documento,

hiperenlaces múltiples, enlaces agrupados, atributos para los enlaces, etc. Creado a través del Lenguaje de enlaces extensible (XLL).

- 10 Exportabilidad a otros formatos de publicación (papel, web, cd-rom, etc.). El documento maestro de la edición electrónica podría ser un documento XML que se integraría en el formato deseado de manera directa.[36]

I.7.4 SOAP

SOAP, siglas de Simple Object Access Protocol, es un protocolo estándar creado por Microsoft, IBM y otros, está actualmente bajo el auspicio de la W3C que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP es uno de los protocolos utilizados en los servicios Web. [37]

A diferencia de DCOM y CORBA, que son binarios, SOAP usa el código fuente en XML, que facilita la eliminación de errores, pero es menos efectivo. El intercambio de mensajes se realiza mediante tecnología de componentes. El término *Object* en el nombre significa que se adhiere al paradigma de la programación orientada a objetos.

SOAP es un marco extensible y descentralizado que permite trabajar sobre múltiples pilas de protocolos de redes informáticas. Los procedimientos de llamadas remotas pueden ser modelados en la forma de varios mensajes SOAP interactuando entre sí. SOAP corre sobre cualquier protocolo de Internet, generalmente HTTP. SOAP tiene como base XML, con un diseño que cumple el patrón Cabecera-Desarrollo de diseño de software, como otros muchos diseños, verbigracia HTML. La cabecera (*header*) es opcional y contiene metadatos sobre enrutamiento (*routing*), seguridad o transacciones. El desarrollo (*body*) contiene la información principal, que se conoce como carga útil (*payload*). La carga útil se acoge a un esquema de XML propio.

Conclusiones

En este capítulo se abordaron aspectos teóricos del tema que se va a analizar, exponiendo los principales conceptos asociados al dominio del mismo. Se describe el contexto donde se enmarca, las características y dificultades que lo acompañan, así

como las metodologías y tecnologías a utilizar.

1. Se propone utilizar la programación orientada a objetos por los beneficios que está brinda, por lo que se considera RUP (Proceso Unificado de Desarrollo) la metodología más apropiada para el desarrollo del proyecto y UML como el lenguaje de modelación necesario en este caso.
2. Debe estar basada en el modelo de programación de 2 capas, que permite el desarrollo independiente del proyecto en cuanto a: Lógica del Negocio y Acceso a Datos.
3. La solución a proponer debe usar la tecnología cliente/servidor para descargar el peso fundamental del proyecto en el servidor de la aplicación.
4. En el caso del servidor de aplicaciones, la opción del Servidor Apache es la idónea, por ser este servidor multiplataforma, gratuito y el más usado en Internet.
5. Una vez que se ha escogido Apache como servidor de aplicación, se selecciona PHP como el lenguaje para desarrollar los módulos necesarios, pues es este quien mejor funciona con el servidor escogido además de que no requiere de pago por su uso.
6. La tecnología de los servicios web permite desarrollar aplicaciones multiplataforma y construir un sistema grande a partir de pequeños módulos con funcionalidades propias.
7. Para el uso de esta tecnología de servicios web de forma fácil y que en un futuro pueda ser escalable se escoge el protocolo SOAP y al mismo tiempo el uso del lenguaje de marcado XML para el intercambio de datos entre módulos. Ambos (SOAP y XML) están estandarizados globalmente en Internet.
8. Dentro de las bibliotecas para PHP disponibles en Internet, una de la más completa es PEAR, contando con una serie de módulos que implementan las tareas más comunes en la ejecución de una aplicación dada. Es decir, que en la mayoría de los casos sólo es necesario reutilizar estos módulos y

añadirles las nuevas funcionalidades de nuestra aplicación. Teniendo en cuenta el presente proyecto, PEAR ya cuenta con módulos para desarrollo de servicio web sobre SOAP, manejo XML, etc...

9. A nivel de la capa de datos, las opciones son varias y cualquiera de ellas resulta beneficiosa, por lo que se decide trabajar en función de lograr un sistema multiplataforma, aunque para el desarrollo concreto del proyecto se escoge MySql.

Capítulo II: “Análisis y Diseño de la Solución”

En este capítulo se realiza un análisis del diseño de la solución, empleando el lenguaje de modelado UML, la metodología RUP con la herramienta de desarrollo Rational Rose, para ello se utilizan artefactos tales como, modelo de dominio, modelo de objetos del dominio, reglas del negocio a considerar, así como el análisis del sistema a proponer. Se realiza un estudio de factibilidad para obtener los costos y beneficios del producto.

II.1 Descripción del Modelo de Dominio

Un modelo del dominio captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las "cosas" que existen o los eventos que suceden en el entorno en el que trabaja el sistema. Muchos de los objetos del dominio o clases pueden obtenerse de una especificación de requisitos o mediante la entrevista con los expertos del dominio. [38]

El objetivo del modelado del dominio es comprender y describir las clases más importantes dentro del contexto del sistema. [39]

Para lograr una mayor comprensión del contexto del sistema se describen los objetos del dominio.

II.1.1 Definición de los objetos y los conceptos principales

En el modelo de dominio se definen las siguientes clases fundamentales:

Estudiante: Es el usuario que utiliza el SE y se registra su comportamiento en el sistema.

Asignatura: Materia que imparte un profesor a un grupo de estudiantes y que utiliza los Softwares Educativos.

Extensiones del Sistema: Softwares que se adicionan al sistema para mejorar su funcionamiento o para incluir nuevas funcionalidades al mismo.

Softwares Educativos: Herramienta utilizada como complemento para las

asignaturas.

II.1.2 Representación del Modelo de Objetos de Dominio.

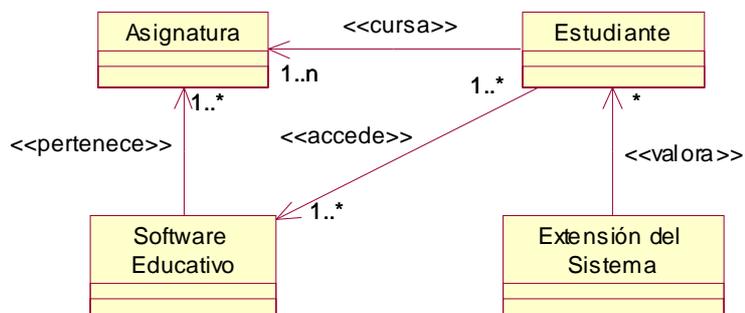


Fig 1 Diagrama de Objetos del Dominio

II.1.3 Reglas del negocio a considerar.

- 1 La plataforma PCUSE podrá ser accedida por: softwares educativos y extensiones del sistema.
- 2 Para que un software educativo pueda registrar datos en PCUSE tiene que contemplar el uso de los servicios y SOAP como protocolo de acceso a los mismos.
- 3 Para hacer uso de PCUSE los usuarios deben estar registrados previamente.
- 4 Los datos de usuario pueden ser modificados en cualquier momento por un usuario con privilegios administrativos.
- 5 Los usuarios del sistema pueden estar activos o inactivos.
- 6 Un profesor puede tener más de una asignatura asignada.
- 7 Un profesor solo tiene acceso a las asignaturas que tiene asignada y a su vez a los SSEE de estas.
- 8 Un profesor puede utilizar más de un SE en cada asignatura.

II.2 Descripción del Sistema Propuesto.

II.2.1 Concepción general del sistema.

El resultado que se pretende alcanzar en esta investigación es una plataforma que garantice el control del uso de los softwares educativos en los centros de educación, por la importancia que ha tomado el uso de softwares como apoyo al proceso docente educativo. Con este software se trata de eliminar el problema descrito en el Capítulo I: Fundamentación Teórica.

El producto brindará posibilidades de almacenar información concerniente al desarrollo de un estudiante en una herramienta educativa, dando la facilidad de ser utilizado por cualquier sistema sin importar la plataforma ni gestor de base de datos en que fue desarrollado. La plataforma será transparente a los usuarios, solo será de interés por los sistemas que la utilicen.

II.2.2 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”. [40]

Los requerimientos funcionales del sistema propuesto son los siguientes:

1. Administración
 - 1.1. Validar Usuario (Profesores y Estudiante)
 - 1.2. Insertar datos de usuario
 - 1.3. Actualizar datos de usuario
 - 1.4. Habilitar usuario
 - 1.5. Inhabilitar usuario
 - 1.6. Recuperar datos de usuario
 - 1.7. Cambio de contraseña
 - 1.8. Asignar permisos a usuario

2. Software Educativo

- 2.1. Insertar datos de software educativo
- 2.2. Mostrar softwares educativos
- 2.3. Actualizar datos de software educativo
- 2.4. Habilitar software educativo
- 2.5. Inhabilitar software educativo
- 2.6. Insertar datos de visitas de estudiantes a software educativo
- 2.7. Insertar datos de visitas a contenido de software educativo
- 3. Asignatura
 - 3.1. Asignar asignaturas a profesor
 - 3.2. Actualizar asignaturas a profesor
 - 3.3. Mostrar asignatura

II.2.3 Requerimientos no Funcionales.

Requerimientos de interfaz externa

1. El acceso a las funcionalidades debe utilizar tipos de datos estándar de Internet.
2. Las funcionalidades deben estar accesibles a través del protocolo SOAP.

Requerimientos de Usabilidad

1. La plataforma no puede ser accedida directamente, sino a través de una interfaz diseñada para estos propósitos.
2. El sistema estará dirigido a desarrolladores de SSEE y extensiones del sistema.
3. Los mensajes de error deben ser reportados por la propia aplicación en la medida de las posibilidades y no por el Sistema Operativo. Los mensajes del sistema deben estar en el idioma apropiado.

Requerimientos de Rendimientos

El sistema deberá responder en el mínimo de tiempo posible ante las solicitudes de información por parte de otros sistemas y en el procesamiento de la información. La eficiencia de la aplicación estará determinada en gran medida por el

aprovechamiento de los recursos que se disponen en el modelo de n capas, y la velocidad de las consultas a la base de datos.

Requerimientos de Soporte

Se documentará la aplicación con un manual de ayuda con el objetivo de explicar el uso de la plataforma para garantizar el soporte de la herramienta. Se debe realizar el proyecto de forma versionable que permita darle mantenimientos al sistema a fin de aumentar las funcionalidades y/o corregir los errores del mismo a través de versiones posteriores. Los servicios de instalación y mantenimiento del sistema será responsabilidad del administrador en la entidad que sea utilizado.

Requerimientos de Seguridad

La información estará protegida contra accesos no autorizados utilizando mecanismos de validación que puedan garantizar el cumplimiento de esto: cuenta, contraseña y nivel de acceso, de manera que cada uno pueda tener disponible solamente las opciones relacionadas con su actividad y tenga datos de acceso propios, garantizando así la confidencialidad.

Se usarán mecanismos de encriptación de los datos que por cuestiones de seguridad no deben viajar al servidor en texto plano, como es el caso de las contraseñas. Se guardará encriptada esta información en la base de datos utilizando para ello MD5 como algoritmo de encriptación.

Requerimientos Políticos

La aplicación debe cumplir con lineamientos, políticos y/o regulaciones de la entidad que utilice el sistema.

Requerimientos de Confiabilidad

El sistema debe ser tolerante ante los fallos y las operaciones a realizar deben ser transaccionales.

Ayuda y documentación en línea

El sistema tendrá un manual de ayuda disponible que permitirá aclarar dudas respecto al funcionamiento del mismo.

Requerimientos de Software

La aplicación debe poderse ejecutar en diferentes entornos, como *Windows, Linux, etc.* (Multiplataforma). Al mismo tiempo debe ser capaz de usar para guardar los datos diferentes motores de bases de datos.

Requerimientos de Hardware

Los requerimientos de hardware estarán dados por la plataforma específica que se utilice para la instalación del sistema, en cuanto a sistema operativo, servidor de aplicaciones y gestor de bases de datos.

II.3 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. El modelo de casos de uso sirve como acuerdo entre clientes y desarrolladores, y proporciona la entrada fundamental para el análisis, el diseño y las pruebas.” [41]

“Un modelo de casos de uso es un modelo del sistema que contiene actores, casos de uso y sus relaciones.”

“El modelo de casos de uso describe lo que hace el sistema para cada tipo de usuario.”

II.3.1 Actores del sistema.

Un actor no es más que un conjunto de roles que los usuarios de Casos de Uso desempeñan cuando interactúan con estos Casos de Uso. Los actores representan a terceros fuera del sistema que colaboran con el mismo. Una vez identificado los actores del sistema, queda identificado el entorno externo del

sistema. [35]

Actor	Justificación
Software Educativo	Actor que almacena en el sistema registros de visitas de la utilización por parte de los estudiantes.
Extensión del Sistemas	Actor que utiliza el sistema para consumir la información almacenada y a partir de esta elaborar reportes.

Tabla 1 Actores del Sistema

II.3.2 Casos de Uso del Sistema.

“La forma en que los actores usan el sistema se representa con un caso de uso. Los casos de uso son “fragmentos” de funcionalidad que el sistema ofrece para aportar un resultado de valor para sus actores. De manera más precisa, un caso de uso especifica una secuencia de acciones que el sistema puede llevar a cabo interactuando con sus actores, incluyendo alternativas dentro de la secuencia.” [42]

Los casos de uso del sistema son los siguientes:

1. Autenticar Estudiante
2. Autenticar Trabajador
3. Insertar datos de usuario.
4. Actualizar datos de usuario.
5. Cambiar estado usuario.
6. Recuperar datos de usuario.
7. Cambio de contraseña.
8. Asignar permisos a usuarios
9. Insertar datos de software educativo.
10. Mostrar softwares educativos
11. Actualizar datos de software educativo.
12. Cambiar estado de software educativo.
13. Insertar datos de visitas de estudiantes a software educativo.
14. Insertar datos de visitas a contenido de software educativo.

- 15. Asignar asignatura a profesor
- 16. Actualizar asignatura a profesor.
- 17. Mostrar asignatura.

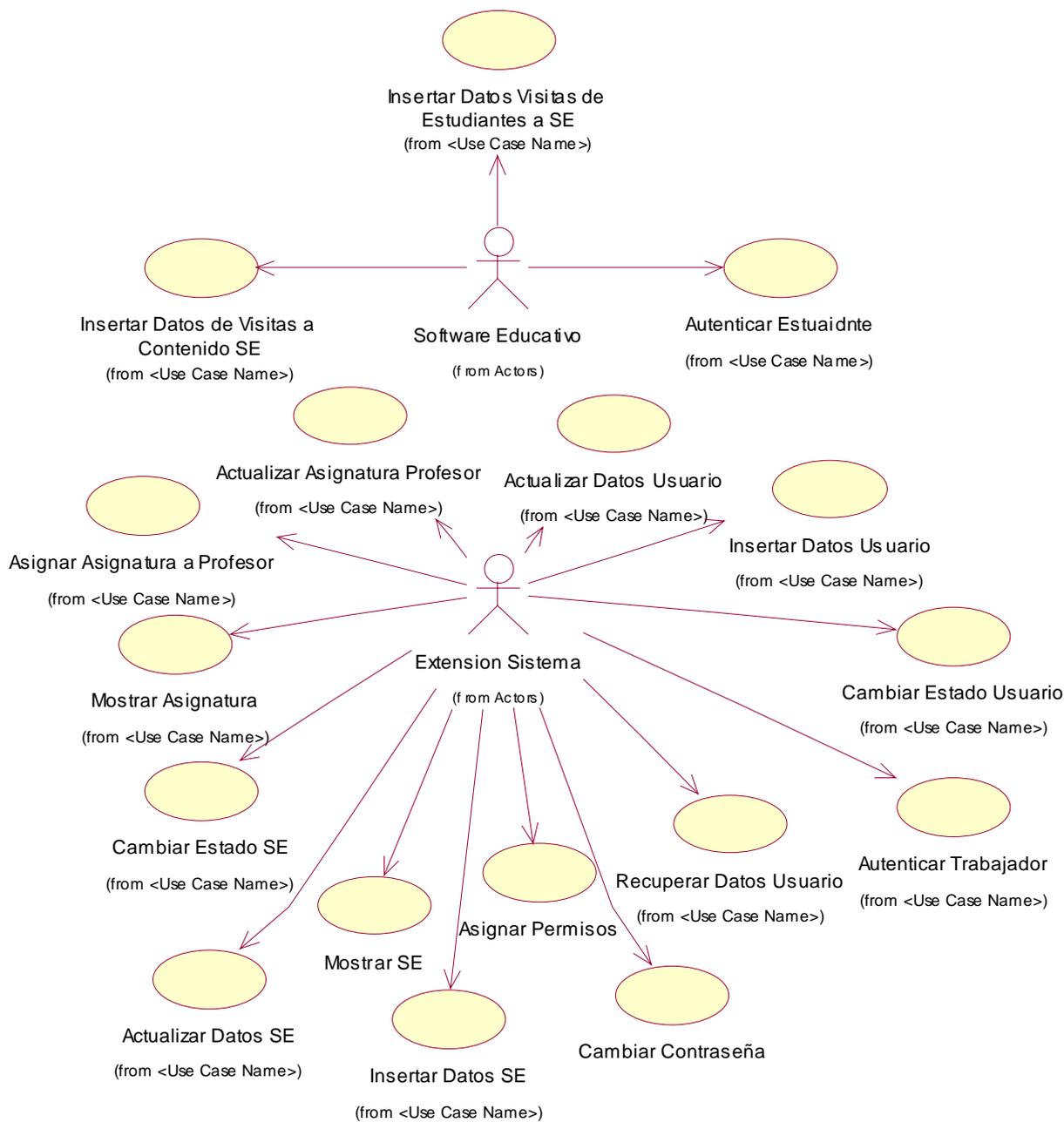


Fig 2 Diagramas de Casos de Uso del Sistema

II.3.3 Descripción de los Casos de Uso del Sistema.

Caso de uso: Autenticar Estudiante
Actores: Software Educativo (inicia).
Propósito: Comenzar a trabajar.
Resumen: El caso de uso se inicia cuando un software educativo envía los datos de inicio e sesión al sistema. El caso de uso culmina con la respuesta del sistema que puede ser afirmativa o negativa
Precondiciones: El SE no puede estar autenticado en el sistema.
Referencias: R 1.1
Poscondiciones: En caso de datos correctos se autentica el software educativo, en caso de un error se deniega el acceso al sistema.

Tabla 2 Descripción del caso de uso de sistema: Autenticar Estudiante

Caso de uso: Autenticar Trabajador
Actores: Extensión del sistema (inicia).
Propósito: Comenzar a trabajar.
Resumen: El caso de uso se inicia cuando un sistema externo envía los datos de inicio e sesión al sistema. El caso de uso culmina con la respuesta del sistema que puede ser afirmativa o negativa
Precondiciones: La extensión del sistema no puede estar autenticada en el sistema.
Referencias: R 1.1

<p>Poscondiciones:</p> <p>En caso de datos correctos se autentica la extensión del sistema, en caso de un error se deniega el acceso al sistema.</p>
--

Tabla 3 Descripción del caso de uso de sistema: Autenticar Trabajador

Caso de uso: Insertar datos de usuario.
Actores: Sistema Externo (inicia).
<p>Propósito:</p> <p>Insertar en el sistemas, nombre, apellidos, carnet de identidad, fecha de nacimiento, cuenta y contraseña de acceso al sistema.</p>
<p>Resumen:</p> <p>El caso de uso se inicia cuando el sistema externo envía los datos para el registro de usuario, y culmina cuando PCUSE devuelve una respuesta que notifica si los datos fueron insertados o no.</p>
<p>Precondiciones:</p> <p>El usuario del que se desea registrar los datos no puede estar registrado.</p>
Referencias: R 1.2
<p>Poscondiciones:</p> <p>Quedan insertados los datos del usuario</p>

Tabla 4 Descripción del caso de uso de sistema: Insertar datos de usuario

Caso de uso: Actualizar datos de usuario.
Actores: Extensión del sistema (inicia).
<p>Propósito</p> <p>Actualizar en el sistemas, nombre, apellidos, carnet de identidad, fecha de nacimiento, cuenta o contraseña de acceso al sistema:</p>

<p>Resumen:</p> <p>El caso de uso se inicia cuando el sistema externo envía los datos para la actualización de los datos de un usuario, y culmina cuando PCUSE devuelve una respuesta que notifica si los datos fueron actualizados o no.</p>
<p>Precondiciones:</p> <p>El usuario del que se desea actualizar los datos debe ser usuario del sistema.</p>
<p>Referencias: R 1.3</p>
<p>Poscondiciones:</p> <p>Quedan actualizados los datos del usuario.</p>

Tabla 5 Descripción del caso de uso de sistema: Actualizar datos de usuario

<p>Caso de uso: Cambiar estado usuario.</p>
<p>Actores: Extensión del sistema (inicia).</p>
<p>Propósito:</p> <p>Habilitar o Inhabilitar un usuario en el sistema.</p>
<p>Resumen:</p> <p>El caso de uso se inicia cuando el sistema externo envía los datos para habilitar o Inhabilitar un usuario en el sistema, y culmina cuando PCUSE devuelve una respuesta que notifica si la operación fue realizada o no.</p>
<p>Precondiciones:</p> <p>El usuario que se desea habilitar o inhabilitar debe ser usuario del sistema.</p>
<p>Referencias: R 1.4 y 1.5</p>
<p>Poscondiciones:</p> <p>Queda habilitado o inhabilitado el usuario en dependencia de la opción</p>

que se pretenda.

Tabla 6 Descripción del caso de uso de sistema: Cambiar estado usuario.

Caso de uso: Recuperar datos de usuario.
Actores: Sistema externo (inicia).
Propósito: Obtener los datos del usuario registrado.
Resumen: El caso de uso se inicia cuando el sistema externo le hace una petición a PCUSE de los datos de un usuario determinado. Finaliza con una respuesta positiva o negativa, en caso positivo devuelve los datos.
Precondiciones:
Referencias: R 1.6
Poscondiciones: Si el usuario existe se devuelven los datos, sino se notifica el error.

Tabla 7 Descripción del caso de uso de sistema: Recuperar datos de usuario.

Caso de uso: Cambio de contraseña.
Actores: Sistema externo (inicia).
Propósito: Cambiar la contraseña a los usuarios.
Resumen: El caso de uso se inicia cuando el sistema externo le envía a PCUSE los datos de un usuario, cuenta y clave de usuario. Finaliza cuando el sistema recibe notificación de la modificación.
Precondiciones: Debe estar registrado el usuario en el sistema
Referencias: R 1.7

Poscondiciones:

Tabla 8 Descripción del caso de uso de sistema: Cambio de contraseña.

Caso de uso: Asignar permisos a usuario.
Actores: Sistema externo (inicia).
Propósito: Asignar permiso a usuarios para el acceso diferentes módulos.
Resumen: El caso de uso se inicia cuando el sistema externo le envía a PCUSE los datos los permisos que le son concedidos a un usuario. Finaliza cuando el sistema recibe notificación de la modificación.
Precondiciones: Debe estar registrado el usuario en el sistema y creados los módulos para otorgar los permisos.
Referencias: R 1.8
Poscondiciones:

Tabla 9 Descripción del caso de uso de sistema: Asignar permisos a usuario

Caso de uso: Insertar datos de software educativo.
Actores: Sistema externo (inicia).
Propósito: Registrar nombre de software educativo en el sistema.
Resumen: El caso de uso se inicia cuando el sistema externo envía nombre de software educativo a PCUSE para que se registre. Finaliza con una respuesta positiva o negativa de la inserción.
Precondiciones: El nombre del SE no puede existir en el sistema
Referencias: R 2.1

<p>Poscondiciones: Queda insertado el SE en el sistema.</p>

Tabla 10 Descripción del caso de uso de sistema: Insertar datos de software educativo

Caso de uso: Mostrar softwares educativos.
Actores: Sistema externo (inicia).
Propósito: Mostrar los softwares educativos existentes en el sistema.
Resumen: El caso de uso se inicia cuando un sistema externo hace una petición a PCUSE para mostrar los softwares educativos existentes en el sistema.
Precondiciones: Debe existir datos de los softwares educativos en el sistema
Referencias: R 2.2
Poscondiciones:

Tabla 11 Descripción del caso de uso de sistema: Mostrar softwares educativos

Caso de uso: Actualizar datos de software educativo.
Actores: Sistema externo (inicia).
Propósito: Actualizar nombre de un SE en el sistema.
Resumen: El caso de uso se inicia cuando el sistema externo envía a PCUSE nombre de software educativo para actualizar un SE ya existente. Finaliza con la actualización..
Precondiciones: El SE tiene que existir en el sistema.

Referencias: R 2.3
Poscondiciones: Queda actualizado el nombre del SE.

Tabla 12 Descripción del caso de uso de sistema: Actualizar datos de software educativo.

Caso de uso: Cambiar estado SE.
Actores: Sistema externo (inicia).
Propósito: Habilitar o Inhabilitar SE en el sistema.
Resumen: El caso de uso comienza cuando se envía a PCUSE la orden de habilitar o inhabilitar un software educativo en el sistema, culminando con el cambio de estado del SE.
Precondiciones: El SE tiene que existir en el sistema.
Referencias: R 2.4 y 2.5
Poscondiciones: Queda habilitado o inhabilitado el SE.

Tabla 13 Descripción del caso de uso de sistema: Cambiar estado SE.

Caso de uso: Insertar datos de visitas de estudiantes a software educativo.
Actores: Software Educativo (inicia).
Propósito: Registrar datos de visita de un estudiante a un SE.
Resumen: El caso de uso comienza cuando un software educativo envía a PCUSE datos referentes a la visita de un estudiante a un SE, estos datos son: nombre estudiante, apellidos, hora de entrada, hora de salida, etc. Culminando con el registro en el sistema de los datos.

Precondiciones:
Referencias: R 2.6
Poscondiciones: Quedan registrados los datos en el sistema.

Tabla 14 Descripción del caso de uso de sistema: Insertar datos de visitas de estudiantes a software

Caso de uso: Insertar datos de visitas a contenido de software educativo.
Actores: Software Educativo (inicia).
Propósito: Registra los datos de los contenidos que un estudiante visitó en un SE.
Resumen: El caso de uso comienza cuando un software educativo envía a PCUSE datos referentes al contenido visitado por un estudiante en un SE, estos datos son: contenido, tiempo en cada contenido, respuestas a las preguntas, si existen, etc. Culminando con el registro en el sistema de los datos.
Precondiciones:
Referencias: R 2.7
Poscondiciones: Quedan registrados los datos en el sistema.

Tabla 15 Descripción del caso de uso de sistema: Insertar datos de visitas a contenido de software.

Caso de uso: Asignar asignatura a profesor
Actores: Sistema Externo (inicia).
Propósito: Registrar asignatura(s) que imparte un profesor

<p>Resumen:</p> <p>El caso de uso se inicia cuando un sistema externo envía a PCUSE la información de la asignatura que va hacer asignada a un profesor. Culminando con el registro en el sistema de los datos.</p>
<p>Precondiciones:</p> <p>Debe existir la asignatura en el sistema.</p>
<p>Referencias: R 3.1</p>
<p>Poscondiciones:</p> <p>Quedan registrados los datos en el sistema.</p>

Tabla 16 Descripción del caso de uso de sistema: Asignar asignatura a profesor

<p>Caso de uso: Actualizar datos asignatura a profesor</p>
<p>Actores: Sistema externo (inicia).</p>
<p>Propósito: Modificar registro de asignatura (s) a un profesor</p>
<p>Resumen:</p> <p>El caso de uso se inicia cuando un sistema externo le hace petición a PCUSE de la asignatura o asignaturas asignadas a un profesor y se modifica el registro de la misma. Culminando con el registro en el sistema de los datos.</p>
<p>Precondiciones:</p> <p>Debe ser seleccionado el profesor para modificar el registro de la asignatura.</p>
<p>Referencias: R 3.2</p>
<p>Poscondiciones: Quedan registrados los datos en el sistema.</p>

Tabla 17 Descripción del caso de uso de sistema: Actualizar asignatura a profesor

Caso de uso: Mostrar asignaturas.
Actores: Sistema externo (inicia).
Propósito: Mostrar todas las asignaturas pertenecientes a un carrera
Resumen: El caso de uso se inicia cuando un sistema externo le hace petición a PCUSE de las asignaturas pertenecientes a una carrera.
Precondiciones: Las asignaturas deben estar registradas en sistema.
Referencias: R 3.3
Poscondiciones:

Tabla 18 Descripción del caso de uso de sistema: Mostrar Asignatura.

II.4 Factibilidad Económica

“Para llevar a cabo un buen proyecto de desarrollo de software, debemos comprender el ámbito del trabajo a realizar, los recursos requeridos, las tareas a ejecutar, las referencias a tener en cuenta, el esfuerzo (COSTE) a emplear y la agenda a seguir”. [43]

Para determinar si la construcción de un software es factible o no se han introducido en la Ingeniería del Software una serie de técnicas utilizadas dentro de las tareas de planificación, que ayudan a planificar y controlar el esfuerzo y el tiempo necesario de desarrollo:

- Técnicas de estimación del esfuerzo (costo) de desarrollo. Dentro de las cuales se sitúa el Modelo Constructivo de Costes (Constructive Cost Model (COCOMO)).
- Técnicas de planificación y seguimiento de proyectos. [44]

“La estimación es una de las primeras actividades de la gestión de proyectos informáticos. Se le define como la predicción del personal, del esfuerzo, de los costos

y del tiempo que se requerirán para realizar todas las actividades y construir todos los productos asociados con el proyecto. Su objetivo es conocer en etapas tempranas y de manera aproximada, el costo, la duración y los recursos necesarios para el desarrollo de proyectos de software. Se trata de una apreciación del futuro y la exactitud con la que ésta se realice, depende la mayoría de las veces de una buena herramienta de estimación, de la experiencia del estimador y del acceso a una base de información histórica de los proyectos.”[45]

II.4.1 Planificación por puntos de función.

Nombre de la entrada externa	Cantidad de ficheros	Cantidad de Elementos de datos	Clasificación (Bajo, Medio y Alto)
Insertar datos de usuario	2	9	Medio
Modificar datos de usuario	2	9	Medio
Buscar usuario	2	8	Medio
Insertar permisos usuario	2	5	Medio
Insertar permisos profesor	2	5	Medio
Insertar datos de asignatura	8	15	Alto
Actualizar datos de asignatura	8	15	Alto
Eliminar asignatura al profesor	1	2	Bajo
Iniciar sesión	4	14	Alto
Modificar contraseña	1	1	Bajo
Modificar contraseña usuario registrado	1	1	Bajo
Insertar datos de SE	2	3	Bajo
Modificar datos de SE	1	2	Bajo
Insertar datos de visitas de estudiantes a software educativo	1	6	Bajo

Insertar datos de visitas a contenido de software educativo	1	8	Bajo
---	---	---	------

Tabla 19 Planificación: Entradas Externas

Nombre de la petición	Cantidad de ficheros	Cantidad de Elementos de datos	Clasificación (Bajo, Medio y Alto)
Obtener carreras	1	3	Bajo
Obtener años por carrera	3	5	Bajo
Obtener asignatura por año	5	7	Alto
Obtener asignatura por profesor	5	8	Alto
Obtener asignatura por carrera	5	8	Alto
Obtener software por asignatura	3	4	Bajo
Obtener si hay pregunta	1	8	Bajo

Tabla 20 Planificación: Peticiones

Nombre del fichero interno	Cantidad de records	Cantidad de Elementos de datos	Clasificación (Bajo, Medio y Alto)
cod_actividadlaboral	1	2	Bajo
cod_cargo	1	2	Bajo
cod_claseestudiante	1	2	Bajo
cod_conceptoingreso	1	2	Bajo
cod_municipio	1	4	Bajo
cod_nacionalidad	1	2	Bajo
cod_niveleducacional	1	2	Bajo
cod_paises	1	2	Bajo
cod_procedenciaescolar	1	2	Bajo
cod_provincia	1	3	Bajo
cod_raza	1	2	Bajo
cod_sexos	1	2	Bajo
cod_situacionescolar	1	2	Bajo
cod_tipoevaluacion	1	2	Bajo
cod_viaingreso	1	2	Bajo
estado	1	2	Bajo
subestado	1	3	Bajo

tb_annos	1	3	Bajo
tb_area	1	4	Bajo
tb_asignatura	1	7	Bajo
tb_asignaturaxmultimedia	1	2	Bajo
tb_asignaturaxprofesor	1	2	Bajo
tb_carreras	1	3	Bajo
tb_contenido	1	5	Bajo
tb_contenidoxestudiante	1	8	Bajo
tb_disciplinas	1	4	Bajo
tb_estudiante	1	3	Bajo
tb_estudiantexmultimedia	1	6	Bajo
tb_grupo	1	4	Bajo
tb_modulos	1	3	Bajo
tb_multimedia	1	2	Bajo
tb_niveles	1	2	Bajo
tb_operadores	1	4	Bajo
tb_operadorxmodulos	1	3	Bajo
tb_persona	1	15	Bajo
tb_planestudio	1	5	Bajo

tb_preasignatura	1	5	Bajo
tb_pregunta	1	3	Bajo
tb_profesor	1	1	Bajo
tb_semestre	1	4	Bajo
tb_sesiones	1	4	Bajo
tb_subarea	1	5	Bajo
tb_trabajador	1	2	Bajo

Tabla 21 Planificación: Ficheros Internos.

Elementos	Bajos	X Peso	Medios	X Peso	Altos	X Peso	Subtotal de puntos de función
Entradas externas	7	3	5	4	3	6	59
Salidas externas	0	4	0	5	0	7	0
Peticiones	4	3	0	4	3	6	30
Ficheros lógicos internos	43	7	0	10	0	15	301
Ficheros de interfaces	0	5	0	7	0	10	0

externas							
Total de Puntos de Función No Ajustados (TPFNA)							390

Tabla 22 Planificación: Puntos de Función

Características		Valor
Puntos de función desajustados		390
Lenguaje	MySQL	PHP
Instrucciones fuentes por puntos de función (SLOC*PFNA)	40	44
Por ciento de la aplicación en cuanto a requerimientos funcionales	40%	60%
Instrucciones fuentes	156*40=6240	234*44=10296
Total de instrucciones fuentes	16 536	

Tabla 23 Planificación: Miles de Instrucciones Fuertes

II.4.2 Determinación de los costos

Determinación de los valores de las variables de costos utilizadas en el cálculo de costos en la producción de software, como el cálculo del esfuerzo, el tiempo de desarrollo, la cantidad de hombres y el costo total del sistema.

Factores de Escala SF _j	Valor	Justificación
Precedentes	3,72	La experiencia previa del equipo de desarrollo en este tipo de proyectos es casi

(PREC)		sin precedentes (Nominal).
Flexibilidad (FLEX)	3,04	Refleja en el proceso de desarrollo flexibilidad entre la relación de las especificaciones de los requerimientos preestablecidos y las de la interfaz externa (Nominal).
Arquitectura/resolución del riesgo (RESL)	4, 24	Al analizar la amplitud de análisis de riesgo del producto a desarrollar y teniendo en cuenta la experiencia de nuestros profesionales en investigaciones de este tipo podemos decir que existen algunos factores de riesgo (Nominal).
Cohesión del Equipo (TEAM)	1,10	La relación entre los miembros del equipo que trabajarán en el desarrollo del producto es altamente cooperativa. (Muy alto)
Madurez del Proceso (PMAT)	6,24	El desarrollo del sistema se encuentra a un Nivel I (Alto), por lo que podemos decir que está en su primera etapa un poco avanzada. (Bajo)

Tabla 24 Costos: Factores de escalas

Factores de Ajuste	Valor	Justificación
Fiabilidad y complejidad del producto (RCPX)	1	El producto será fiable y tendrá una moderada complejidad pues estará basada en la Web y la base de datos (Nominal)

		BD moderada, no se requiere de amplia documentación. La aplicación Web tiene una moderada complejidad. (Nominal)
Requerimientos de reusabilidad (RUSE)	1,07	Se utilizara código reusable en toda la aplicación (Alto).
Dificultad de la plataforma (PDIF)	1	La plataforma de aplicación tiene gran estabilidad. No tiene grandes restricciones en cuanto al tiempo de ejecución ya que el software podrá estar trabajando varias horas. EL Software no tiene limitación de memoria impuesta (Nominal).
Capacidad del personal (PERS)	0,83	El personal dispone de una alta capacidad.
Experiencia del personal (PREX)	0,87	El equipo que desarrollará el software posee un buen dominio y conocimiento del lenguaje de programación y el diseño de base de datos. Con una experiencia de aproximadamente algunos años (Alto).
Facilidades para el desarrollo (FCIL)	0,87	Para el desarrollo del software se utilizarán en el diseño, la herramienta CASE Rational Rose 2003 utilizando UML como lenguaje de modelado y para la programación del mismo PHP como lenguaje, como SGBD se escogió MYSQL, servidor de aplicaciones web Apache, la utilización de estas herramientas facilitará el desarrollo del software (Alto).

Esfuerzo de calendario (SCED)	1	Para el desarrollo del producto se ha realizado una planificación del esfuerzo de calendario alto, por parte del equipo que desarrollará el sistema (Alto).
-------------------------------	---	---

Tabla 25 Costos: Factores de Ajuste

Multiplicador de esfuerzos

$$EM = \prod_{i=1} E_{mi} = RCPX * RUSE * PDIF * PERS * PREX * FCIL * SCED$$

i=1

$$EM = 1 * 1,07 * 1 * 0,83 * 0,87 * 0,87 * 1 = 0,67$$

Factores de Escala

$$SF_j = \sum SF_j = PREC + FLEX + RESL + TEAM + PMAT$$

$$SF_j = 3,72 + 3,04 + 4,24 + 1,10 + 6,24 = 18,34$$

Valores de los coeficientes

$$A = 2,94; B = 0,91; C = 3,67; D = 0,24$$

$$E = B + 0,01 * SF_j$$

$$F = D + 0,2 * (E - B)$$

$$E = 0,91 + 0,01 * 18,34$$

$$F = 0,24 + 0,2 * (1,0934 - 0,91)$$

$$E = 1,0934$$

$$F = 0,27668$$

Esfuerzo

$$PM = A * (MF)^E * EM$$

Donde: MF es miles de instrucciones fuentes

$$MF = 16,536 \approx 16,5$$

$$PM = 2,94 * (16,5)^{1,0934} * 0,67$$

$$PM = 2,94 * 21,43 * 0,67$$

$$PM = 42,21$$

Cálculo del tiempo de desarrollo

$$TDEV = C * PM^F$$

$$TDEV = 3,67 * (42,21)^{0,27668}$$

$$TDEV = 10,33 \approx 10 \text{ meses}$$

Cálculo de la cantidad de hombres

$$CH = PM / TDEV$$

$$CH = 42,21 / 10,33$$

$$CH = 4,08 \approx 4 \text{ hombres}$$

Costo

Se asume como salario promedio mensual \$ 225

$$CHM = 4 * \text{Salario Promedio}$$

$$CHM = 4 * 225$$

$$CHM = 900 \text{ \$/mes}$$

$$\text{Costo} = CHM * PM$$

$$\text{Costo} = \$ 900 * 42,21$$

$$\text{Costo} = \$ 37 989$$

II.4.3 Beneficios tangibles e intangibles

Los beneficios obtenidos con el desarrollo del software son fundamentalmente intangibles, pues permite al profesor tener un control del uso de los Softwares Educativos por parte de los estudiantes y poder emitir un criterio evaluativo para cada uno de ellos además posibilita realizar un tratamiento individualizado atendiendo a las especificidades de cada estudiante. Por otra parte el sistema estimulará su uso ya que se diseñó con tecnologías de punta teniendo en cuenta el empleo de las nuevas tecnologías de informática y las comunicaciones aplicadas a la educación.

II.4.4 Análisis de costos y beneficios

Al desarrollo de todo producto informático va asociado a un costo, el justificarlo depende de los beneficios tangibles e intangibles que produce. En el caso de la aplicación a desarrollar se llegó a la conclusión que se logra aproximadamente en 10 meses con una fuerza de trabajo de 4 hombres con salario promedio de \$225 pesos por trabajador para un costo total de \$ 37 989.

II.4.5 Tratamiento de Excepciones.

La técnica para el manejo de los errores en el sistema se concebirá de manera que cuando ocurra un error se genere una excepción; es decir, la ejecución normal se detenga y se transfiera el control a la zona de tratamiento de excepciones.

Los mensajes de error que emita el sistema ya sea de la base de datos o del servidor de aplicación se traducirán a un lenguaje comprensible para el usuario.

II.5 Modelo de Clases Web.

Un diagrama de clases Web representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase. [41] Al tratar de utilizar el diagrama de clases tradicionales 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.

Caso de Uso	Diagrama de Clases
Autenticar Estudiante	Anexo 1
Autenticar Trabajador	Anexo 2
Insertar datos de usuario.	Anexo 3
Actualizar datos de usuario.	Anexo 4
Cambiar estado usuario.	Anexo 5
Recuperar datos de usuario.	Anexo 6
Cambio de contraseña	Anexo 7
Asignar permisos a usuario	Anexo 8
Insertar datos de software educativo.	Anexo 9
Mostrar softwares educativos	Anexo 10
Actualizar datos de software educativo.	Anexo 11
Cambiar estado de software educativo.	Anexo 12.
Insertar datos de visitas de estudiantes a software educativo.	Anexo 13
Insertar datos de visitas a contenido de software educativo.	Anexo 14
Asignar asignatura a profesor	Anexo 15
Actualizar asignatura a profesor.	Anexo 16
Mostrar asignaturas	Anexo 17

Tabla 26 Diagramas de Clases Web.

II.6 Diseño de la Base de datos.

En este epígrafe se muestra el diseño de la base de datos del sistema propuesto a través de los diagramas de clases persistente y el esquema de la base de datos

generados a partir de este, con el modelo de datos.

II.6.1 Diagrama del modelo lógico de datos.

El diagrama de clases persistentes muestra todas las clases capaces de mantener su valor en el espacio y en el tiempo [46]. Se muestra en el Anexo17

II.6.2 Diagrama del modelo físico de datos.

El modelo de datos que muestra la estructura física de las tablas de la base de datos, obtenido a partir del diagrama de clases persistentes [47], es mostrado en el Anexo18.

II.7 Diagrama 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 como dependen los componentes unos de otros. [48]

La figura 4 muestra el diagrama de implementación.

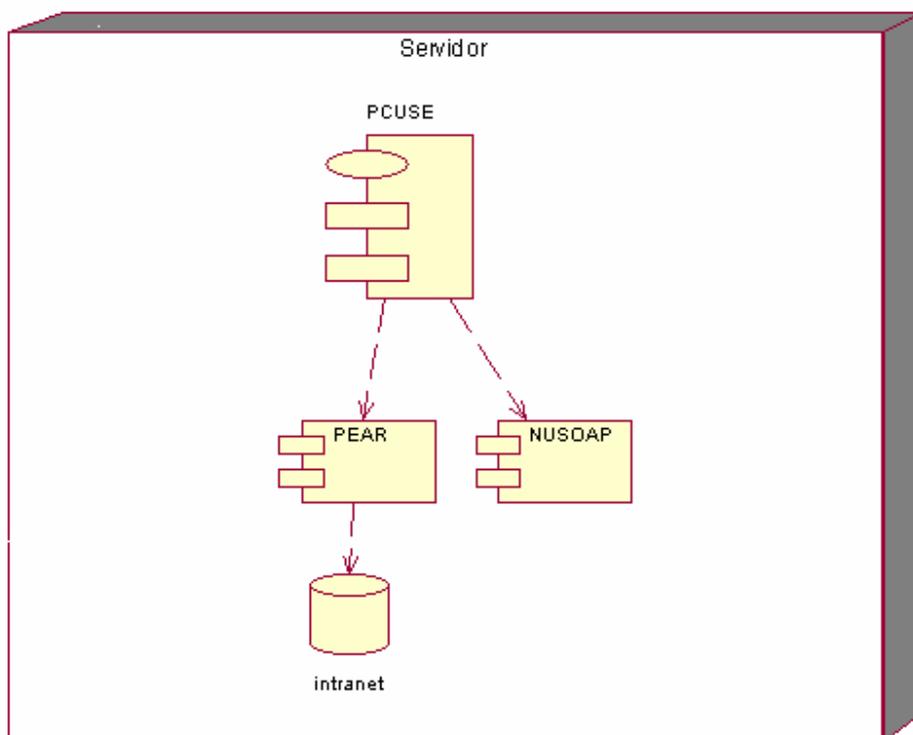


Fig 3 Diagrama de Implementación

Conclusiones

En este capítulo se presentó la solución propuesta del sistema, utilizando el modelo del dominio donde se definen los diferentes conceptos referente a este y sus relaciones, lo que llevó a lograr una mejor comprensión del problema a resolver y permitió con una mayor claridad hacer mención a los requerimientos funcionales y no funcionales, así como a los casos de uso con su respectiva descripción.

Se calcularon además, los beneficios tangibles e intangibles de la solución y los costos que requiere.

Se muestran los diagramas de clases de la aplicación realizada, se describe el diseño de la base de datos y el diagrama de implementación de la aplicación, como RUP propone.

Capítulo III: “Análisis de los Resultados”

La validación de los resultados es la etapa final de todo proyecto, pues es la encargada de verificar que los requerimientos solicitados por el usuario sean cumplidos por la aplicación. Esta contribuye a darle el acabado al proyecto, logrando así la calidad que se requiere.

En este capítulo se analizan los resultados obtenidos al aplicar la Plataforma de Control del Uso de Softwares Educativos (PCUSE).

III.1 Descripción General del Software

La Plataforma está basada en las nuevas tecnologías de la información y las comunicaciones, usando el paradigma de los servicios web que son entidades programables que proporcionan alguna funcionalidad determinada, y son accesibles a cualquier número de sistemas que usen las normas de Internet.

La aplicación consiste en una plataforma que contiene funciones para su consumo. Estas funciones dan la posibilidad a los desarrolladores de software educativos de registrar información referente al comportamiento de un estudiante en el mismo y a desarrollar extensiones del sistema que permitan hacer un análisis de la información recopilada.

En la etapa de diseño se realizaron las especificaciones que contemplan el tratamiento de errores en caso de fallo del sistema.

La aplicación cuenta con una descripción de las funciones en formato HTML que pueden ser consultadas a través de cualquier navegador de Internet proporcionando de esta manera una ayuda para su uso.

III.1.1 Validación

Según la Real Academia de la Lengua Española: validación es acción y efecto de validar, firmeza, fuerza, seguridad o subsistencia de algún acto.

La validación es una etapa que generalmente no se realiza, y la cual es útil para tener la posibilidad de retroalimentación y a su vez facilitar la mejora del sistema, por medio de ella se pueden identificar puntos débiles y fuertes de todo sistema implantado.

El software que se propone solo ha tenido la posibilidad de ser validado un sistema externo y un tutorial que utiliza las funciones de dicha plataforma. Es por ello que se dice que no ha podido ser validado en toda su magnitud en tiempo y carga.

Para realizar la validación se analizaron diferentes métodos, y por las características se decidió aplicar una encuesta, pues la misma constituye la única forma posible de poder recoger la opinión de un gran número de personas para conocer varios aspectos del sistema. El cuestionario aplicado fue cerrado, pues este limita las respuestas posibles a través de un estilo cuidadoso en la pregunta.

III.1.2 Cuestionario

El cuestionario (ver anexo 20) está orientado a conocer el grado de aceptación de la plataforma, la importancia que se le atribuye así como la necesidad de su uso. Cuenta con 4 preguntas:

- 1 La pregunta 1 está dirigida a conocer que piensan los usuarios acerca de la funcionalidad del sistema.
- 2 La pregunta 2 está dirigida a evaluar la manera de mostrar los datos del sistema.
- 3 La pregunta 3 está dirigida a evaluar la posibilidad de recuperar los errores brindados por el sistema.
- 4 La pregunta 4 está dirigida a conocer la aplicabilidad de la ayuda brindada.

Para aplicar la encuesta, se utilizó el Muestreo Aleatorio Simple o al Azar y se tuvo en cuenta la siguiente población para determinar el tamaño de la muestra.

La población está formada por estudiantes de tercer año de la carrera de ingeniería

informática.

III.1.3 Muestreo Aleatorio Simple

El muestreo aleatorio simple es la forma más común para obtener una muestra representativa es la selección al azar (aleatoria). [49]

$$n = \frac{Npq}{(N-1)D + pq} \quad (1)$$

Donde:

n tamaño de la muestra.

p proporción de elementos que cumplen la condición.

q = 1- p proporción de elementos que no cumplen la condición.

D = B²/4 donde B = Error dado por el investigador.

N Tamaño de la población.

Fijando un error B= 0.1

Se prueba que si p = q =0.5 entonces se obtiene n_{max}

$$n = \frac{86 * 0.5 * 0.5}{(85) * 0.0025 + 0.5 * 0.5} \quad (2)$$

$$n = 46 \quad (3)$$

Entonces, de una población de 86 personas, se toma una muestra de 46.

Para hacer el análisis de los datos se utilizó el paquete estadístico SPSS del cual obtuvimos los siguientes datos:

Nota aclaratoria:

El valor 1 significa respuesta afirmativa, el valor 2 significa respuesta negativa.

Pregunta 1

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
Válidos 1,00	46	100,0	100,0	100,0

Pregunta 1

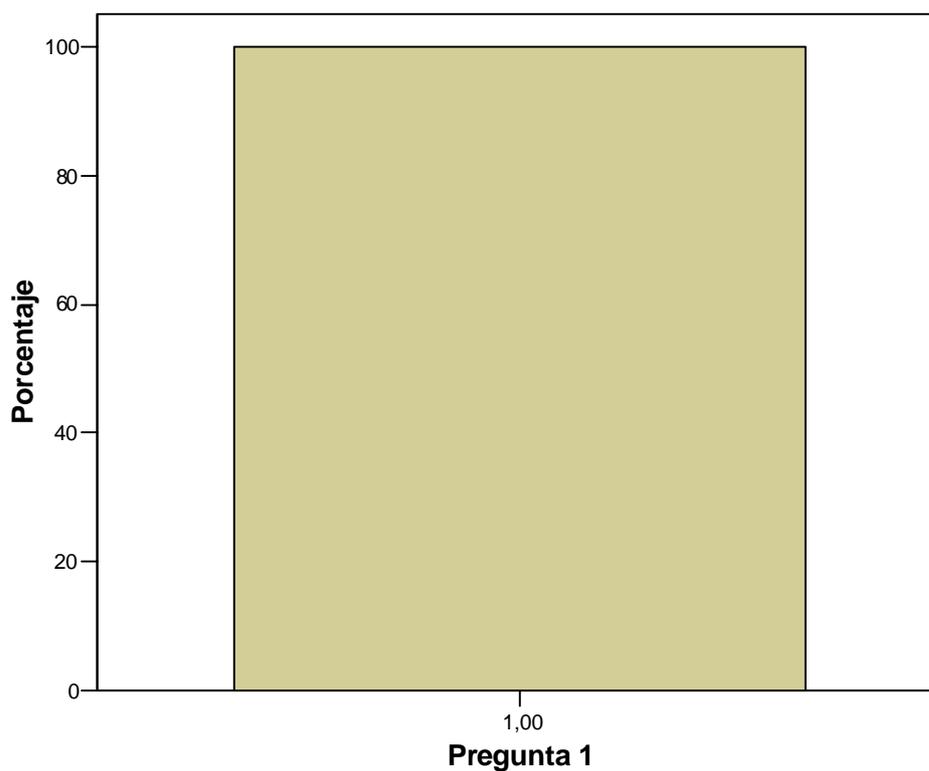


Fig 4 Cuestionario. Pregunta 1

Al aplicar esta pregunta se observó que el 100% de los encuestados están de acuerdo en que es sistema es potencialmente funcional.

Pregunta 2

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
1,00	35	76,1	76,1	76,1
2,00	11	23,9	23,9	100,0
Total	46	100,0	100,0	

Pregunta 2

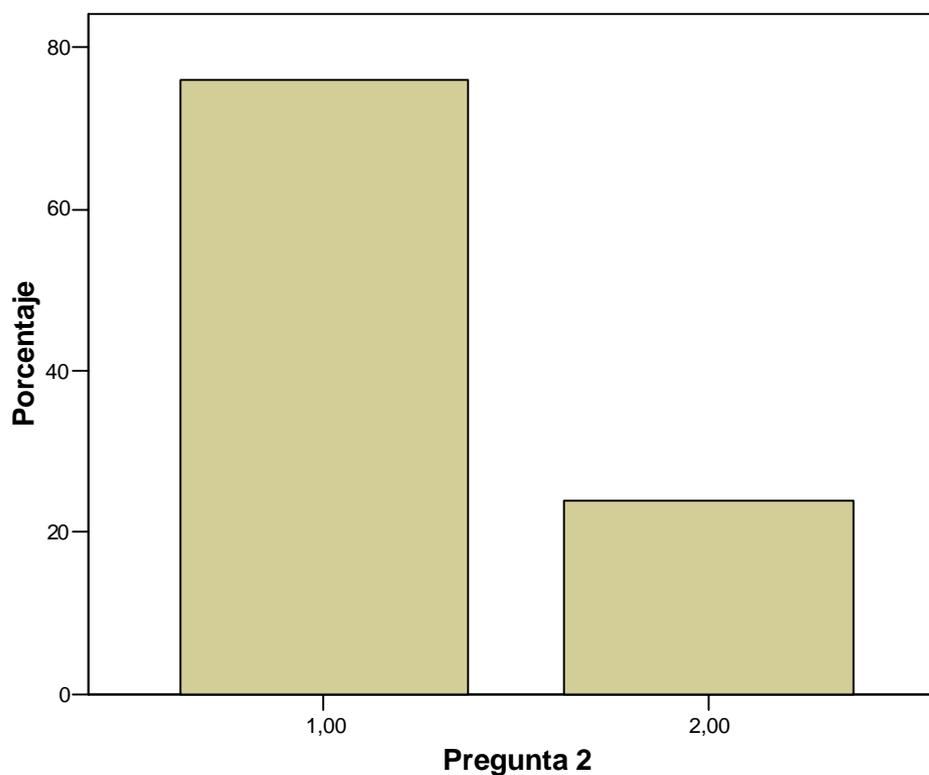


Fig 5 Cuestionario. Pregunta 2

Al aplicar esta pregunta se observa que el 76.1% piensa que el sistema muestra los datos de manera correcta mientras que el 23.9% piensa lo contrario.

Pregunta 3

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
1,00	40	87,0	87,0	87,0
2,00	6	13,0	13,0	100,0
Total	46	100,0	100,0	

Pregunta 3

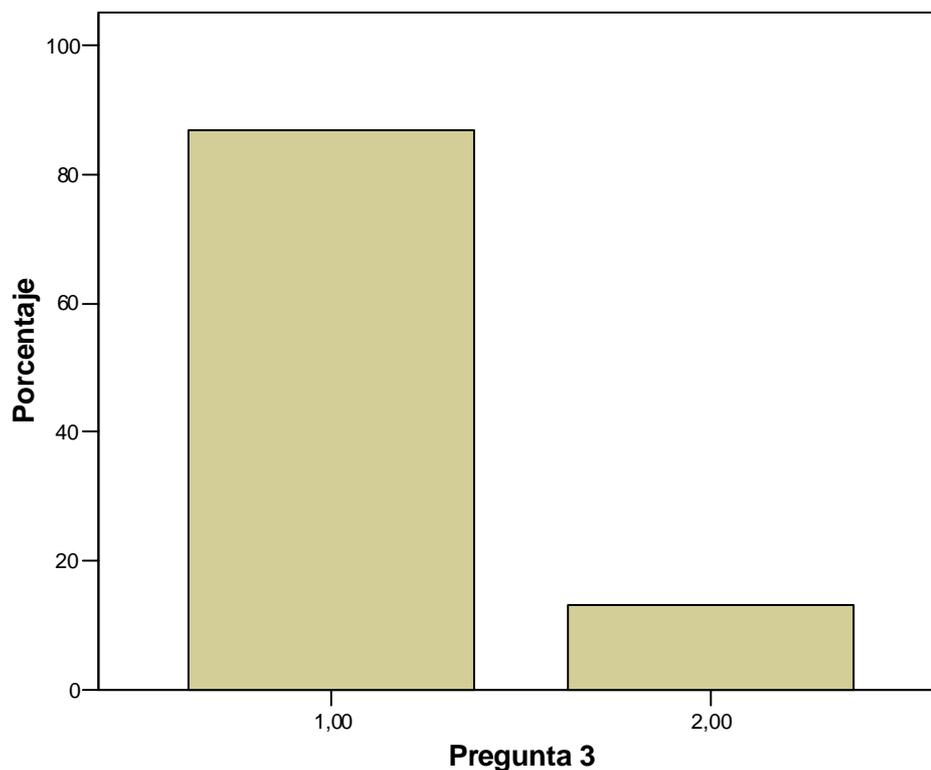


Fig 6 Cuestionario. Pregunta 3

Al aplicar la pregunta 3 se obtiene que el 87% de los encuestados piensan que es buena la manera en que el sistema da la posibilidad de recuperar los errores, el 13 restante no lo considera así.

Pregunta 4

	Frecuencia	Porcentaje	Porcentaje válido	Porcentaje acumulado
1,00	29	63,0	63,0	63,0
2,00	17	37,0	37,0	100,0
Total	46	100,0	100,0	

Pregunta 4

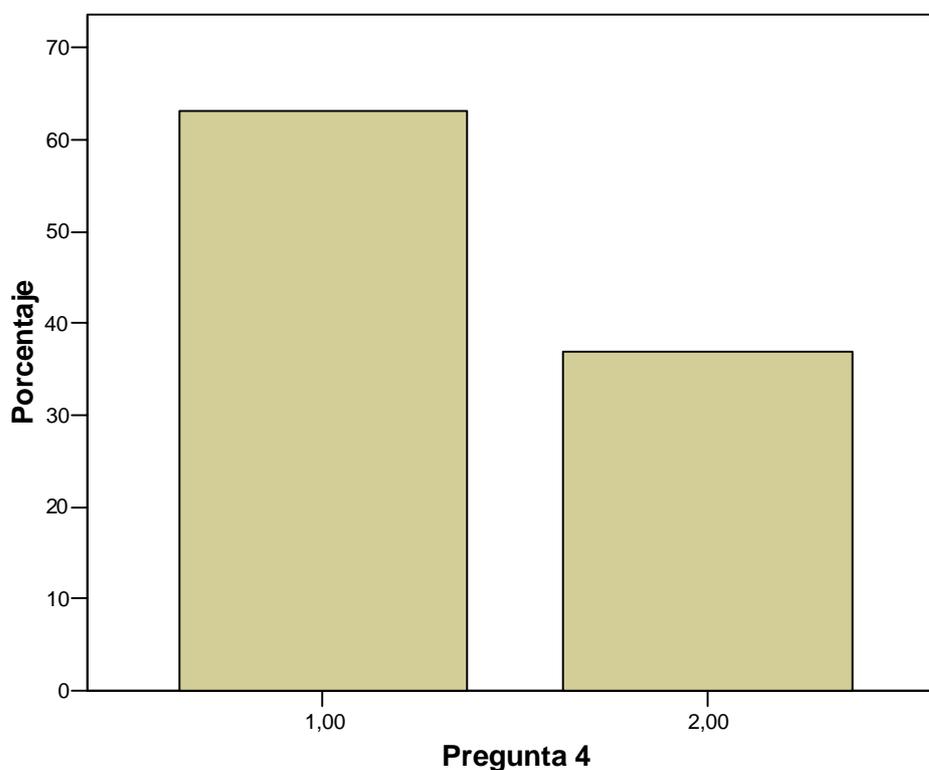


Fig 7 Cuestionario. Pregunta 4

El 63% de los encuestados piensan que la ayuda de PCUSE es buena, mientras que el 37% no están de acuerdo.

Conclusiones

En este capítulo se hizo un análisis de la validación del sistema aplicando una encuesta, se aplicó el método aleatorio simple, para determinar el tamaño de la muestra se utilizó la fórmula (1) y se procesaron los datos en SPSS.

Se llegó a la conclusión de que el sistema en sentido general cumple con las necesidades de los usuarios.

Conclusiones

La inexistencia de un entorno que permita almacenar datos de un software educativo, sin tener en cuenta la plataforma ni lenguaje de programación en que fue desarrollado, motivó al desarrollo de un sistema para el control y uso de los softwares educativos. Esta propuesta permite a los desarrolladores de SSEE utilizar una plataforma que permita guardar datos relacionados con el mismo.

Se realizó un estudio en detalles del uso de los Softwares Educativos por parte de los estudiantes, así como de las necesidades informativas de los profesores, permitiendo esto llegar a determinar los requisitos necesarios para este tipo de aplicación.

El sistema se desarrolló independiente de plataforma lo que puede ser utilizado en cualquier centro educativo. Se siguió la metodología RUP para el desarrollo del proyecto, y se utilizó UML para la modelación de todas las fases del mismo.

La aplicación está provista de un tutorial que muestra el uso de las funciones implementadas en la plataforma.

Para el desarrollo del proyecto se utilizó el paradigma de la Programación Orientada a Objeto con el lenguaje PHP.

Por todo lo anterior se concluye que los objetivos propuestos para el presente proyecto han sido cumplidos.

Recomendaciones

A pesar de que los objetivos del trabajo fueron cumplidos se recomienda:

- 1 Aplicar el sistema en otros centros de educación.
- 2 Realizar una validación con mayor profundidad y en más tiempo con el objetivo de evaluar el funcionamiento.
- 3 Incorporar la plataforma al desarrollo de futuros softwares educativos.
- 4 Mantener la idea de que el sistema debe ser desarrollado sobre la base de la multiplataforma, lo que permitirá adaptarse a las nuevas tecnologías de la informática moderna.

Referencias Bibliográficas

- [1] Colectivo de Autores; Pedagogía. Editorial Pueblo y Educación. La Habana, 2004. pp182-301
- [2] Babanski Yu. K.; Optimización del proceso docente. Editorial Pueblo y Educación. La Habana, 2003.
- [3] Balboa Roberto; Newton_T: un sistema de producción de entrenadores y tutores inteligentes. Memorias del Congreso Iberoamericano de Informática Educativa. Tomo III, Junio, 2004.
- [4] González Castro Vicente; Teoría y práctica de los medios de enseñanza. Editorial Pueblo y Educación. La Habana, 1986.
- [5] Balboa Roberto; Newton_T: un sistema de producción de entrenadores y tutores inteligentes. Memorias del Congreso Iberoamericano de Informática Educativa. Tomo III, Junio, 2004.
- [6] González Castro Vicente; Teoría y práctica de los medios de enseñanza. Editorial Pueblo y Educación. La Habana, 2003.
- [7] García de la Vega Dalia; Pronóstico y Optimización en el diseño de un sistema de cómputo para el CES. Tesis de doctorado. Instituto Superior Politécnico José A. Echevarría. La Habana, 2003.
- [8] Pérez Fernández, V.; Tutoriales para la enseñanza del sistema operativo MS-DOS y las partes fundamentales de una computadora. Tesis de maestría. 2004.
- [9] Alessi Stephen M. y Stanley R. Trollip; Computer-based Instruction. Methods and Development. Prentice-Hall. Inc. Englewood Clifs. New Jersey, 2005.

- [10] Garay Miguel; Las computadoras en la enseñanza superior. Conferencia impartida en el curso de Maestría en Informática Educativa. ISPJAE, La Habana, Junio, 1994.
- [11] García de la Vega Dalia; Pronóstico y Optimización en el diseño de un sistema de cómputo para el CES. Tesis de doctorado. Instituto Superior Politécnico José A. Echevarría. La Habana, 2005.
- [12] Pérez Fernández, V.; Tutoriales para la enseñanza del sistema operativo MS-DOS y las partes fundamentales de una computadora. Tesis de maestría. 2004.
- [13] Pérez Fernández, V.; Tutoriales para la enseñanza del sistema operativo MS-DOS y las partes fundamentales de una computadora. Tesis de maestría. 2004.
- [14] Alessi Stephen M. y Stanley R. Trollip; Computer-based Instruction. Methods and Development. Prentice-Hall. Inc. Englewood Clifs. New Jersey, 2005.
- [15] Chadwick, C.: Educación y Computadoras. En: Nuevas tecnologías de la información y las comunicaciones en la enseñanza, Aique Grupo Editor S.A; Argentina,
- [16] O'Shea Tim y Jhon Self; Enseñanza y aprendizaje con ordenadores. Inteligencia Artificial en Educación. Editorial Científico-técnico. La Habana, 2005.
- [17] Chadwick, C.: Educación y Computadoras. En: Nuevas tecnologías de la información y las comunicaciones en la enseñanza, Aique Grupo Editor S.A; Argentina, 2003
- [18] O'Shea Tim y Jhon Self; Enseñanza y aprendizaje con ordenadores. Inteligencia Artificial en Educación. Editorial Científico-técnico. La Habana, 2000.
- [19] Garay Miguel; Las computadoras en la enseñanza superior. Conferencia impartida en el curso de Maestría en Informática Educativa. ISPJAE, La Habana, Junio, 2004.

[20] Ibidem

[21] Ibidem

[22] Ibidem

[23] Ibídem

[24] Tejada Fernández, José. *"Nuevas tecnologías y educación: consideraciones psicopedagógicas de selección, diseño y aplicación"*. Comunicación y Pedagogía, 2003.

[25] Ibídem

[26] Ibídem

[27] Ibídem

[28] Díaz Antón, Maria Gabriela. *Propuesta de una metodología de desarrollo de software educativo bajo un enfoque de calidad sistemática*. <http://www.academia-interactiva.com/ise.pdf>. (5/5/2007)

[29] Ibídem

[30] Sosa D, Hector K, *Desarrollo de un prototipo de intranet para una Facultad de un Centro de Educación Superior: Modulo Docente*. Trabajo de diploma para optar por el título de Ingeniero Informático. Universidad Carlos Rafael Rodríguez, Cienfuegos, junio 2006.

[31] Ibídem

[32] Ibídem

[33] Ibídem

[34] Ibídem

[35] Ibídem

[36] Ibídem

[37] Ibídem

[38] Jacobson, I. *El Proceso Unificado de Desarrollo de software*. Addison-Wesley, EE.UU., 2000.

[39] Ibídem

[40] Ibídem

[41] Ibídem

[42] Ibídem

[43] Ibídem COCOMO v2 *Modelo de Estimación de Costes para proyectos software.*

[44] Ovejero, Jose Daniel. *Estimación de proyectos para sistemas basados en conocimiento.* Universidad de Cienfuegos, Cienfuegos, 2006.

[45] Ibídem

[46] Jacobson, I. *El Proceso Unificado de Desarrollo de software.* Addison-Wesley, EE.UU., 2000.

[47] Ibídem

[48] Ibídem

[49] Cortés, Manolo Cortes Libro de Generalidades sobre la metodología de la investigación, Universidad Autónoma del Carmen, Campeche México, 2005.

Bibliografías

[ACO 02] Acosta Duarte, Dionel A. *Registros Médicos Pediátricos*. Tesis presentada en opción al título de Master en Informática, Villa Clara, 2002.

[ARO 04] Aronow, D.B. *Information technology applications in quality assurance and quality improvement*, The Joint Commision Journal on Quality Improvement, Vol 19, No 9, EE.UU, Enero 2004.

[BEN 03] Bennatan, E.M. *Software Project Management: A Practitioner's Approach*. Editorial McGraw Hill, EE.UU, 2003.

[BOE 00] Boehm, B. *Software Engineering Economics*. Prentice Hall, EE.UU, 2000.

[BOO 06] Booch, Grady. *Análisis y Diseño Orientado a Objetos*. Ed. Addison-Wesley, EE.UU ,2006.

[DIA 00] Díaz Ferrera, Carlos N. *Sistema de Almacenamiento de la Información*. Editorial ENPES, Ciudad de la Habana, 2000.

[FAI 96] Fairley, Richard. *Ingeniería del Software*. Mc Graw-Hill, Mexico, 1996.

[FRO 01] Frost. *Bases de datos y sistemas expertos*, Bases de datos. Editorial "Evelio Rodríguez Curbelo", La Habana, 2001.

[JON 03] Jones, C. *Programming Productivity*. McGraw Hill, EE UU., 2003.

[PRE 05] Pressman, Roger. *Ingeniería de Software, un enfoque práctico*. Mc Graw-Hill, EE.UU., 2005.

[RUM 00] Rumbaugh, J. *El Lenguaje Unificado de Modelado*. Manual de Referencia. Pearson Education, Madrid, 2000.

[TAY 04] Taylor, E.S. *An Interim Report on Engineering Design*. Massachusetts Institute of Technology, EE.UU., Cambridge, 2004

[ULL 05] Ullman, J.D. *"Principles of Database Systems"*. Computer Science Press, EE.UU., 2005.

[VIZ 03] Vizcarro, C. *Nuevas tecnologías para el aprendizaje*. Pirámide, Madrid, 2003.

[TEJ 04] Tejada Fernández, José. *"Informática e innovación educativa"*. Comunicación Educativa y Nuevas Tecnologías, Praxis, Barcelona, 2004.

[TEJ 1999] Tejada Fernández, José. *"Nuevas tecnologías y educación: consideraciones psicopedagógicas de selección, diseño y aplicación"*. Comunicación y Pedagogía, Praxis, Barcelona, 1999.

[SIM 98] Simon, J. *La educación y la informatización de la sociedad*. Nancea, Madrid, 1998.

[SEV 04] Sevillano, María Luisa. *Nuevas Tecnologías, medios de comunicación y educación*, CCS, Madrid, 2004.

[SAN 04] Sancho Gil, Joana M^a. *Educación en la era de la información*. Revista Cuadernos de Pedagogía, Vol 253, No 253, Marzo, 2004.

[ROI 02] Roig, Rosabel. *Las Nuevas Tecnologías aplicadas a la educación*. Elementos para una articulación didáctica de las Tecnologías de la Información y la Comunicación. Alcoy, Marfil, 2002.

[ROD 03] Rodríguez Diéguez, José Luis. *Nuevas Tecnologías para la Educación* Alfar, Sevilla, 2003.

[PÉR 05] Pérez Pérez, R. *Nuevas tecnologías y nuevos modelos de enseñanza*. CCS, Madrid, 2005.

[GUT 04] Gutierrez Martin, Alfonso. *El profesor ante las nuevas tecnologías multimedia*. Comunicación y Pedagogía, Vol 2, No 153, 2004.

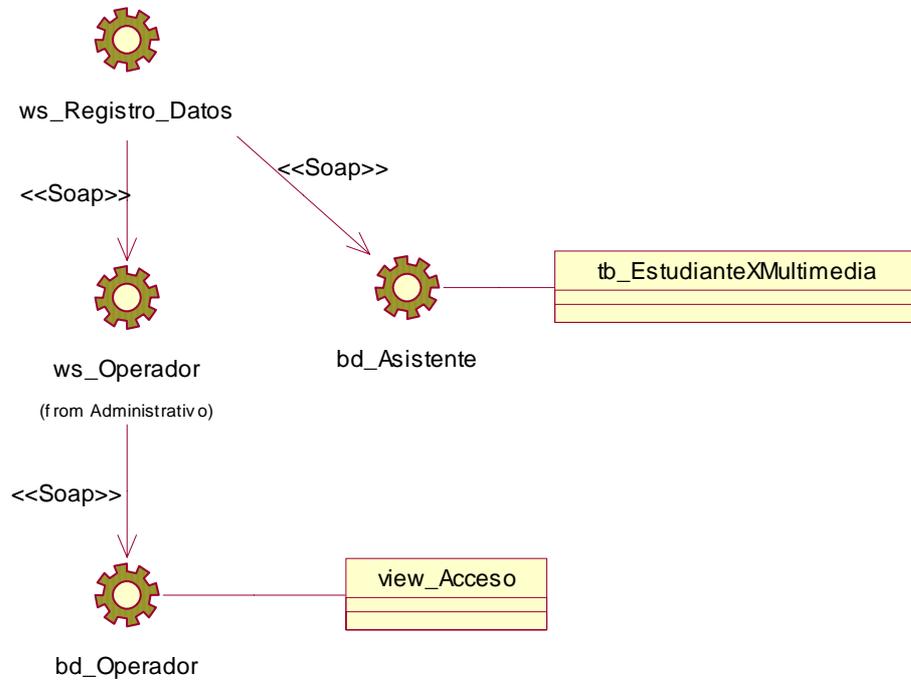
[FER 03] Fernández Muñoz, Ricardo. *Nuevas tecnologías, educación y sociedad*. Editorial CCS, Madrid, 2003.

Conceptos y evolución de la ingeniería del software <http://mondragon.angeltowns.net>. (10/01/07).

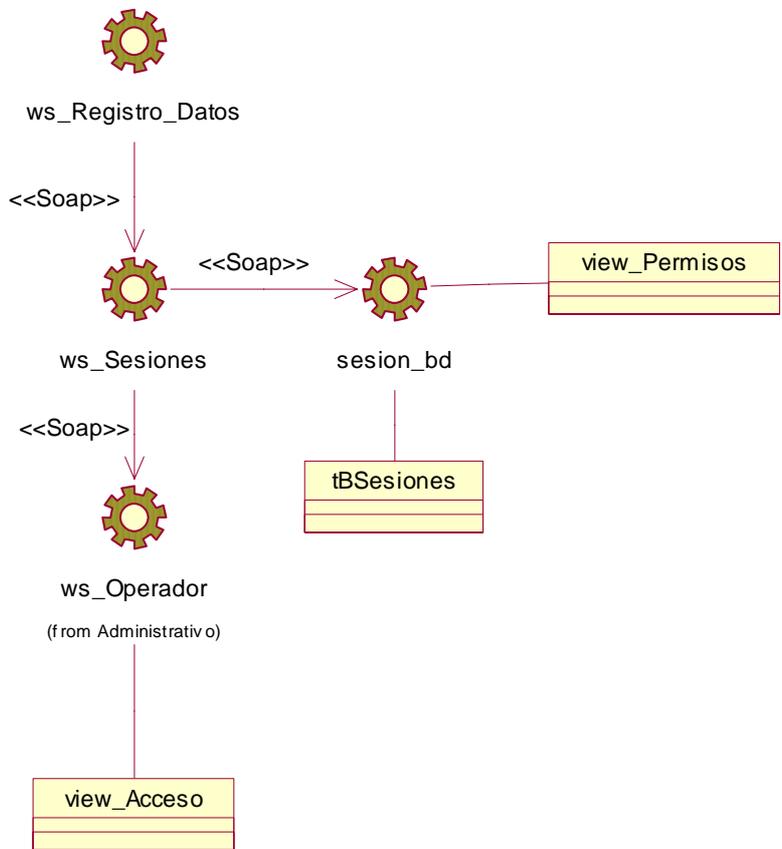
Ramos, Mirtha. *Metodología y criterio para su elaboración y evaluación*. <http://www.uned.ac.cr>. (10/01/07)

Anexos

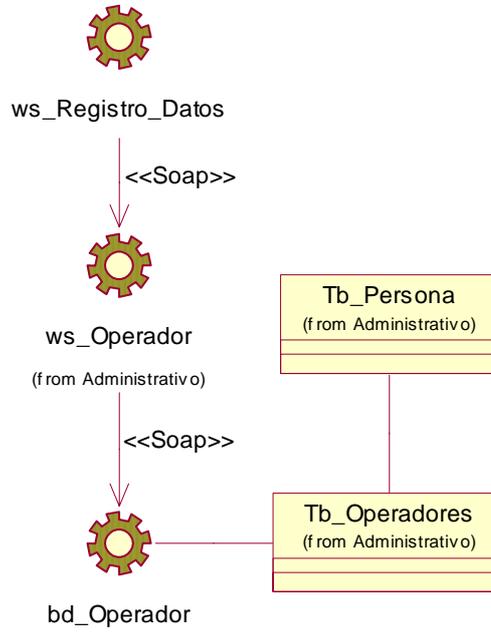
Anexo 1 Diagrama de Clases Web: Autenticar Estudiante



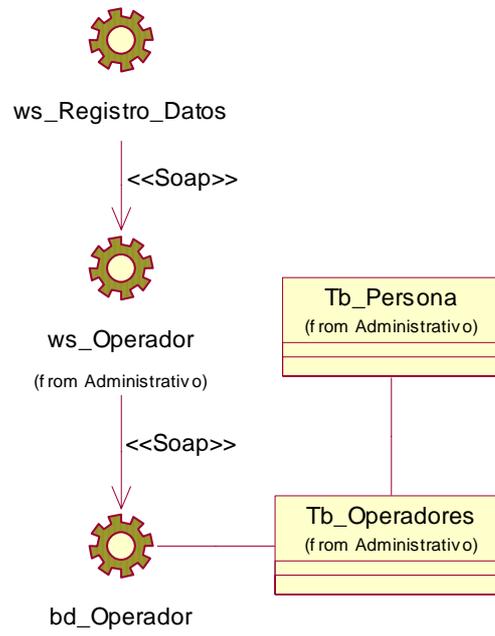
Anexo 2 Diagrama de Clases Web: Autenticar Trabajador



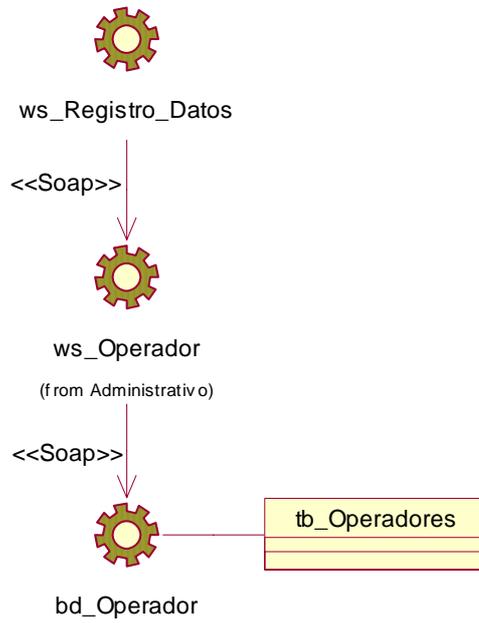
Anexo 3 Diagrama de Clases Web: Insertar datos de usuario.



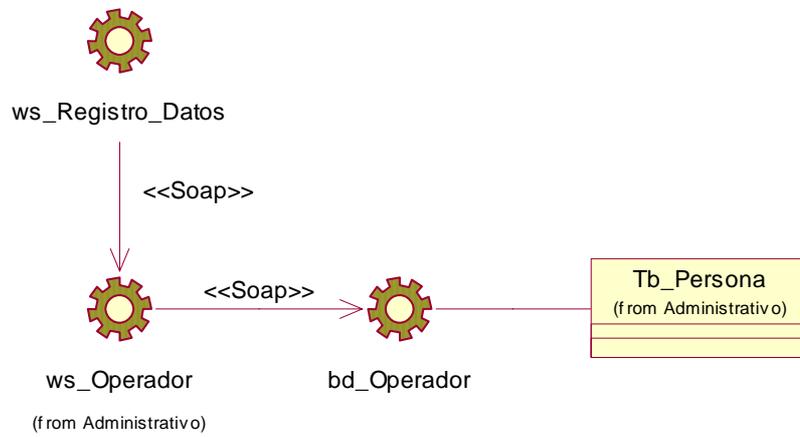
Anexo 4 Diagrama de Clases Web: Actualizar datos de usuario.



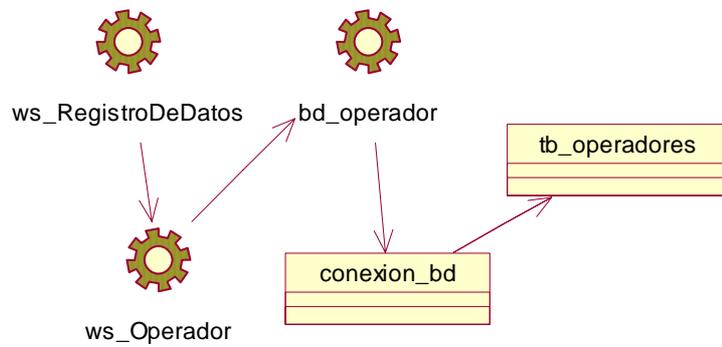
Anexo 5 Diagrama de Clases Web: Cambiar estado usuario.



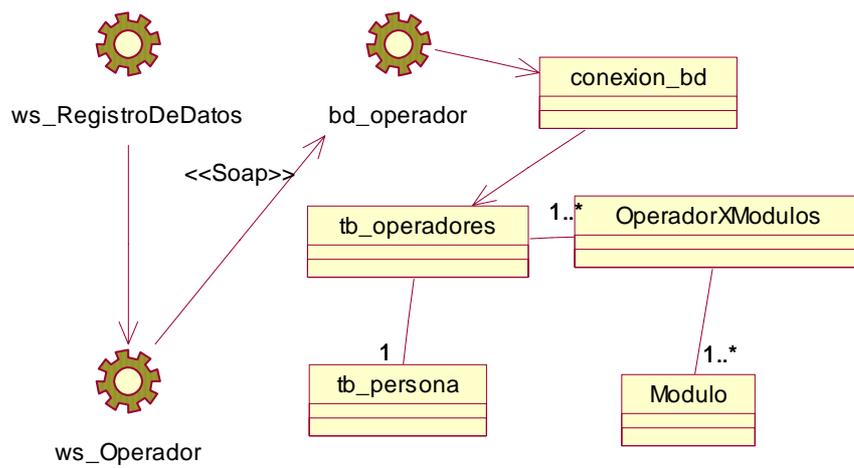
Anexo 6 Diagrama de Clases Web: Recuperar datos de usuario.



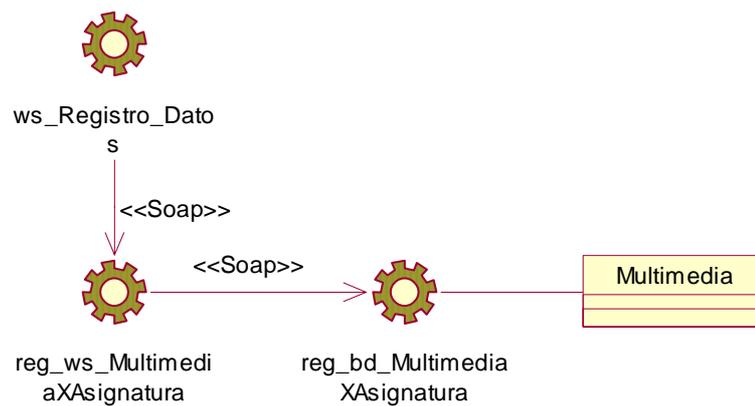
Anexo 7 Diagrama de Clases Web: Cambio de contraseña



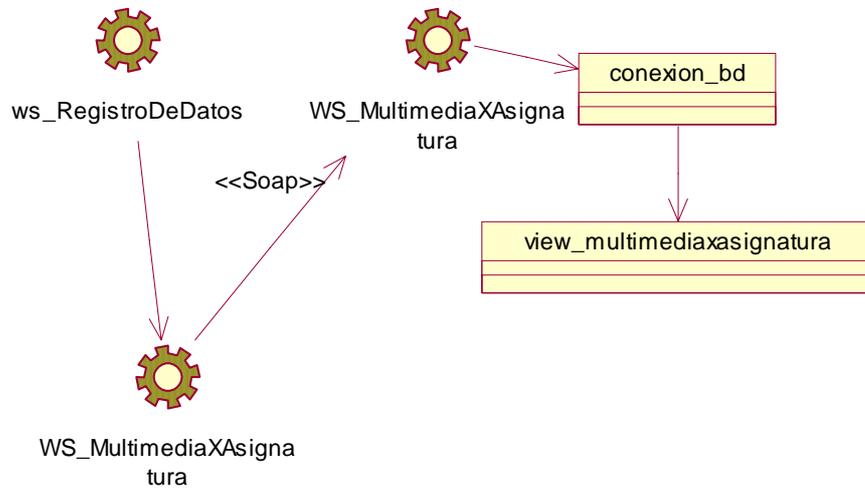
Anexo 8 Diagrama de Clases Web: Asignar permisos a usuario



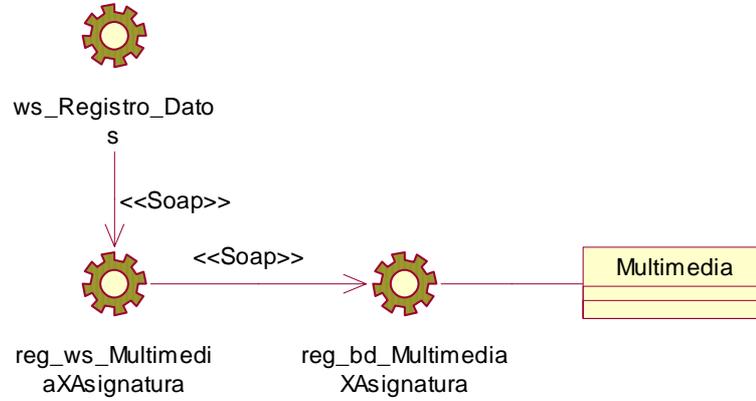
Anexo 9 Diagrama de Clases Web: Insertar datos de software educativo.



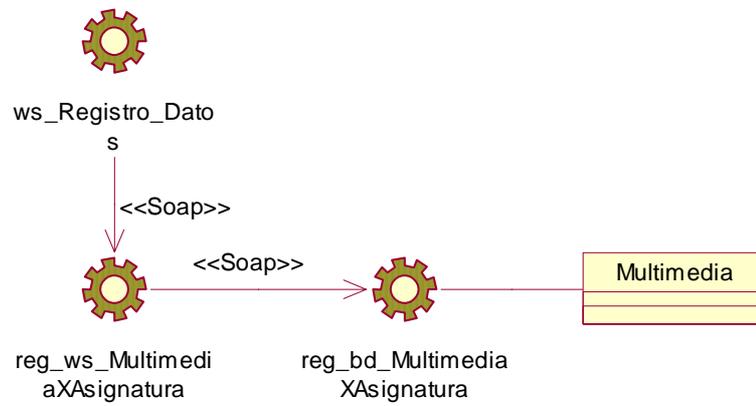
Anexo 10 Diagrama de Clase Web: Mostrar Softwares Educativos.



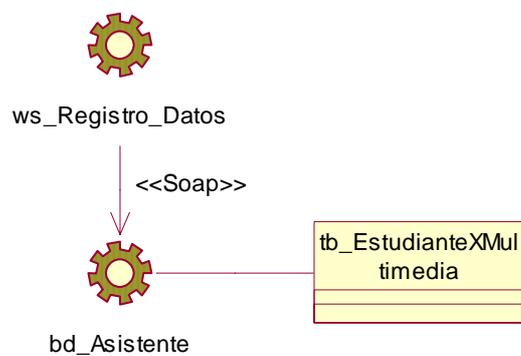
Anexo 11 Diagrama de Clases Web: Actualizar datos de software educativo.



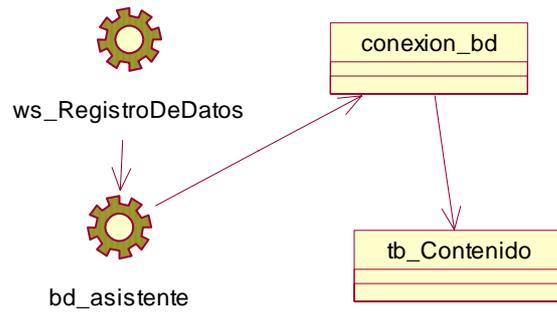
Anexo 12 Diagrama de Clases Web: Cambiar estado de software educativo.



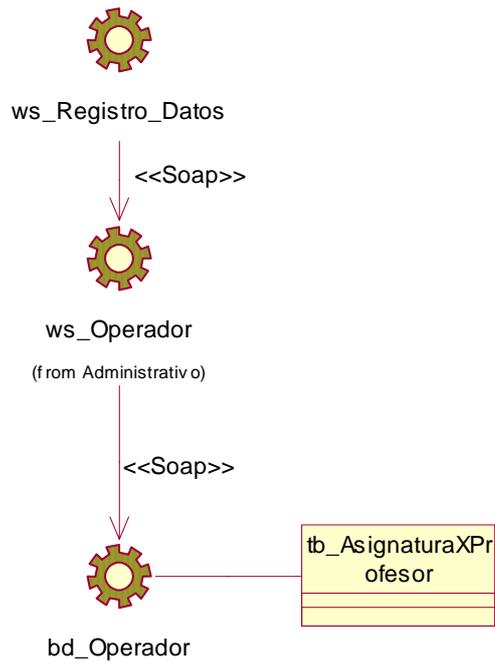
Anexo 13 Diagrama de Clases Web: Insertar datos de visitas de estudiantes a software educativo.



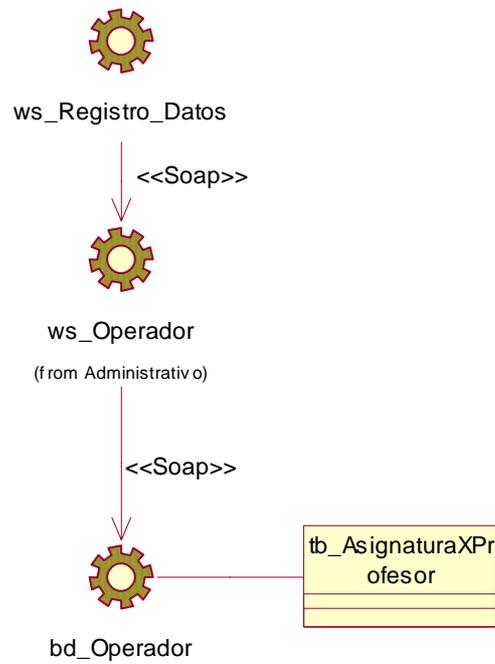
Anexo 14 Diagrama de Clases Web: Insertar datos de visitas a contenidos de Softwares Educativos



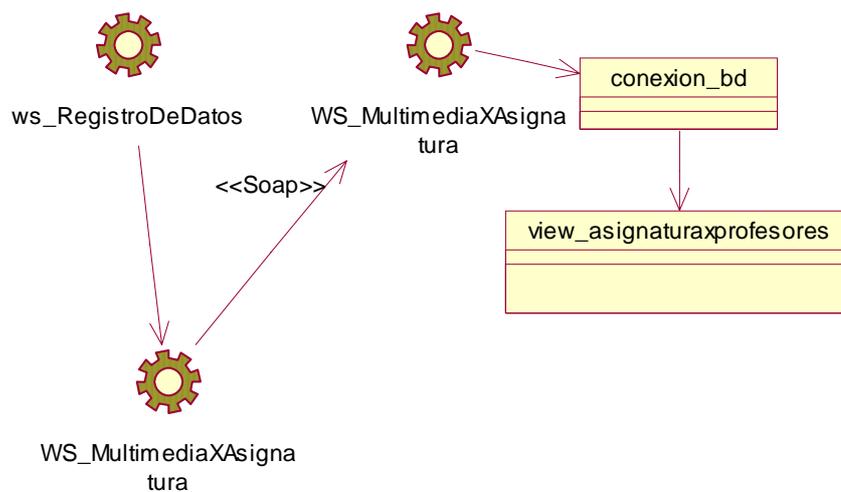
Anexo 15 Diagrama de Clases Web: Asignar Asignatura a Profesor asignatura a profesor



Anexo 16 Diagrama de Clase Web: Actualizar asignatura a profesor



Anexo 17 Diagrama de Clases Web: Mostrar asignaturas



Anexo 17 Diagrama Lógico de Datos

Anexo 18 Diagrama Físico de Datos

Anexo 19 Encuesta

Usted ha sido seleccionado para conocer, en su opinión, el grado de aceptación del entorno.

Cuestionario

- 1) ¿Cree usted funcional la información manejada por la plataforma?
_____ Si _____ No

- 2) ¿La manera en que se brinda las funcionalidades del software permite? ser
_____ Si _____ No

- 3) ¿Está usted conforme con la posibilidad de recuperación de errores que brinda la aplicación?
_____ Si _____ No

- 4) ¿La ayuda que brinda la aplicación es suficiente para su uso?
_____ Si _____ No

Gracias

