



Universidad de las Ciencias Informáticas

Facultad 5

Título: Sistema informático de apoyo al aprendizaje para los estudiantes en la asignatura de Física - Ley de Gauss.

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

Autor: Yoandys Hernández Álvarez.

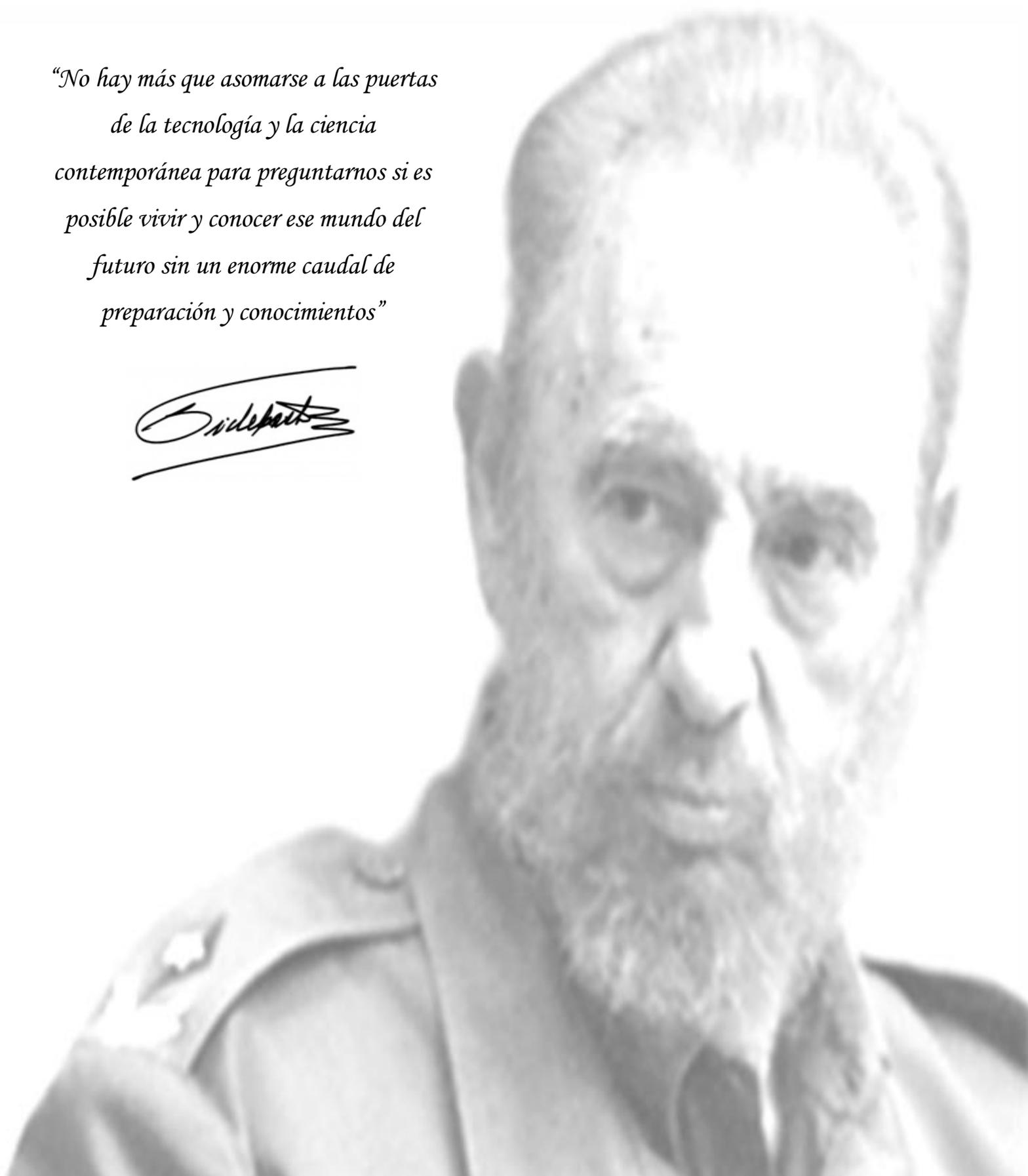
Tutores: Ing. Brenda Batista Fons.

Co-Tutora: Ing. Mayliuvis Esquijarroza Valdés

Ciudad de la Habana, junio de 2012
Año 54 de la Revolución

*“No hay más que asomarse a las puertas
de la tecnología y la ciencia
contemporánea para preguntarnos si es
posible vivir y conocer ese mundo del
futuro sin un enorme caudal de
preparación y conocimientos”*

S. Videla



Declaración de autoría

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

Autor: Yoandys Hernández Álvarez

Tutora: Ing. Brenda Batista Fons

Co-Tutora: Ing. Mayliuvis Esquijarroza Valdés

Datos de contacto

Generales de la Tutora:

Nombre y apellidos: Ing. Brenda Batista Fons.

Especialidad: Ingeniería en Ciencias Informáticas.

Síntesis del(los) tutor(es):

- Graduada en el año 2008.
- 4 años como profesora de Física.
- Actual jefa de la asignatura de Física en la Facultad 5.
- 7 años de experiencia como analista de software.
- Analista principal del proyecto Laboratorios Virtuales.

Correo electrónico: bbatista@uci.cu

Teléfono de contacto: 835-2703.

Generales de la Co-Tutora:

Nombre y apellidos: Ing. Mayliuvis Esquijarrosa Valdés

Especialidad: Ingeniería en Ciencias Informáticas.

Síntesis del(los) tutor(es):

- Graduada en el año 2011.
- 4 años de experiencia como analista de software.
- Profesora de la Universidad de las Ciencias Informáticas.
- Analista principal de un proyecto Nacional.

Correo electrónico: mesquijarrosa@uci.cu

Teléfono de contacto: 837 2135

Agradecimientos

Primero que todo a agradecer a mis padres, Belkis y Luis las personas más grandes que tengo en este mundo, por estar conmigo en todo momento y sobre todo por darme siempre un apoyo incondicional, a los dos les debo todo lo que hoy soy, gracias por dedicar su vida a que a mis hermanos y a mí nos salga todo bien.

A mis hermanos Yurlenis y Yorlandys, no solo por ser los únicos sino por ser los mejores hermanos del mundo, siempre pendientes de mí y sobre todo por quererme mucho, a ellos gracias por existir.

A toda mi familia por confiar en que si podía, brindándome siempre todo su apoyo.

Un agradecimiento muy especial a la persona que fue mi brazo derecho durante 7 años en mi vida y los primeros 4 años en esta universidad; Anaidy. Gracias por estar ahí siempre que te necesitaba y brindarme todo tu cariño. Sabes que eres especial y lo que pasamos juntos nunca lo olvidaremos.

A mi tutora y amiga Brenda por dedicar tantas noches a que yo pudiera ser una mejor persona, por ayudarme tanto en esta universidad y sobre todo por estar pendiente de mis cosas, de mi tesis, como si fuera su hijo, simplemente por ser mí amiga.

A Leyannis por ser la mejor persona del mundo, mi hermana, mi amiga, por todo los que vivimos juntos durante estos 5 años, por ser la persona que hoy es. Por todas las noches sin dormir, hablando de cualquier cosa, por hacerme sentir que lo único que nunca se pierde es un buen amigo.

A amigas Yanet y Annabel por yo ser su pinareño lindo, por ayudarme mucho durante la carrera, por aguantarme tantas noche sin dejarlas dormir. A Richel por ser mi hermano y enseñarme muchas cosas para la vida.

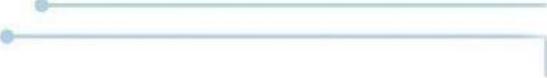
A mi otra tutora y amiga Mayliuvis por siempre estar pendiente de mis cosas y ayudarme durante el desarrollo de este trabajo de diploma.

A mi amigo Yidian por ser un ejemplo para mí de amigo, hermano y guía. Muchas de las cosas que me han hecho madurar tanto en lo personal como profesional se lo debo a sus consejos, por ser mi segundo padre aquí en la UCI.

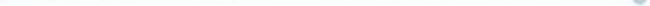
Agradecer enormemente a Luis Arza y Yeily por toda la ayuda brindada en el desarrollo del trabajo de diploma, simplemente por ser excelentes personas.

Agradecer a demás a mis compañeros de cuarto Osmay y Pedro. A los muchachos del 5501, a Diana, Yiselita, Yusleidy, Javier (cacharro), Nayibi, Livan, Noly, Yamilka, Belkis, Yisel Nerys, Yusmila, la profe Zoraida y al secretariado de la FEU de la facultad.

A Fidel y la Revolución por darme la oportunidad de hacer realidad este sueño.



Dedicatoria



Dedicar este sueño que hoy se convierte en realidad a mis padres que siempre me han apoyado y me han dado fuerzas para llegar hasta aquí, a toda mi familia, a mis mejores amigos: Anaidy, Leyannis, Luis Alberto, Yosniel (el pirri), Isbel, Miguel Ángel, José Carlos, Mazola y Marcos.

En fin a todos los que de una forma u otra me guiaron durante la carrera.

Resumen

En los últimos años se ha venido experimentando un acelerado avance de las Tecnologías de la Información y las Comunicaciones (TIC); nuestro país no se encuentra ajeno a este proceso, y dado el elevado volumen de información que se genera actualmente en las entidades cubanas su incorporación en la sociedad es de vital importancia. Actualmente en la esfera educacional se ha reflejado un auge extraordinario en el uso de herramientas informáticas y entornos virtuales de aprendizaje a distancia, los cuales amplían el horizonte de posibilidades dotando tanto a profesores como a estudiantes de la oportunidad de intercambiar en espacios no convencionales o tradicionales, pero igualmente válidos para desarrollar el proceso de enseñanza-aprendizaje de forma exitosa.

La Universidad de las Ciencias Informáticas (UCI), es uno de los centros que promueve el uso de las TIC, además propone un nuevo modelo de formación basado en el auto aprendizaje por parte de los estudiantes, que tiene como base la utilización de diversas herramientas informáticas.

La presente investigación tiene como objetivo principal desarrollar una aplicación web para fortalecer la preparación individual de los estudiantes en el tema de Ley de Gauss y sustentar la estrategia de evaluación seguida por los profesores en la asignatura de Física, basado en el nuevo modelo educacional propuesto.

Para realizar el diseño de la misma se utilizó la metodología “Proceso Unificado de Desarrollo de Software” (RUP), UML (Unified Modeling Language) como lenguaje de modelado y Visual Paradigm como herramienta CASE. Además de NetBeans como entorno de desarrollo, PostgreSQL como Sistema Gestor de Base de Datos, como servidor Web Apache y PHP como lenguaje de programación. Para el diseño propuesto se utilizaron los patrones de diseño Grasp y MVC (Modelo, Vista, Controlador) como estilo arquitectónico.

PALABRAS CLAVES: Aprendizaje, entornos virtuales de aprendizaje y Ley de Gauss.

Índice de Contenidos

| | |
|--|-----------|
| Introducción | 1 |
| CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA | 5 |
| 1.1 Introducción. | 5 |
| 1.2 Proceso de enseñanza aprendizaje (PEA)..... | 5 |
| 1.3 Proceso de enseñanza aprendizaje de la Física en el nivel superior. | 6 |
| 1.4 La utilización de las TIC en la educación..... | 8 |
| 1.5 Definición de tareas, problemas y ejercicios. | 9 |
| 1.6 Sistemas informáticos de apoyo al aprendizaje de la Física existentes en Cuba y el mundo..... | 11 |
| 1.6.1 EDUFI en Energía. | 11 |
| 1.6.2 C@mpus Virtual. | 11 |
| 1.6.3 Gauss View. | 12 |
| 1.6.4 Gauss 09W. | 12 |
| 1.6.5 Entornos Virtuales de Aprendizaje..... | 12 |
| 1.7 Metodologías, herramientas y lenguajes de modelado a utilizar..... | 13 |
| 1.7.1 Metodología de desarrollo de software..... | 13 |
| 1.8 Lenguajes de modelado: | 15 |
| 1.8.1 Lenguaje Unificado de Modelado (UML)..... | 15 |
| 1.8.2 Business Process Modeling Notation (BPMN)..... | 16 |
| 1.9 Herramientas CASE. | 16 |
| 1.9.1 Visual Paradigm. | 16 |
| 1.9.2 Rational Rose Enterprise..... | 17 |
| 1.10 Sistema de Gestión de Bases de Datos (SGBD)..... | 18 |
| 1.10.1 PostgreSQL 8.3..... | 18 |
| 1.10.2 MySQL..... | 19 |
| 1.11 Servidor Web..... | 19 |
| 1.12 Lenguajes de programación para aplicaciones web. | 20 |
| 1.12.1 Lenguajes de programación del lado del cliente..... | 20 |
| 1.13 Lenguaje programación del lado del servidor. | 22 |
| 1.13.1 PHP..... | 22 |
| 1.14 Entorno de desarrollo (NetBeans IDE 6.9)..... | 22 |
| 1.15 Conclusiones parciales. | 23 |

| | |
|---|-----------|
| CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA | 24 |
| 1.1 Introducción. | 24 |
| 1.2 Descripción de la solución propuesta..... | 24 |
| 1.3 Modelo de dominio. | 25 |
| 1.3.1 Diagrama de clases del dominio. | 26 |
| 1.4 Especificación de los requisitos didácticos y de software. | 26 |
| 1.4.1 Requerimientos Funcionales. | 26 |
| 1.4.2 Requerimientos no Funcionales..... | 28 |
| 1.4.3 Requerimientos didácticos..... | 30 |
| 1.5 Modelo de casos de uso del sistema. | 30 |
| 1.5.1 Actores del sistema. | 30 |
| 1.5.2 Diagrama de casos de uso del sistema:..... | 32 |
| 1.5.3 Descripción de los casos de uso del sistema..... | 32 |
| 1.6 Conclusiones parciales..... | 38 |
| CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA. | 39 |
| 3.1 Introducción | 39 |
| 3.2 Descripción de la arquitectura. | 39 |
| 3.2.1 Modelo Vista Controlador (MVC). | 39 |
| 3.3 Patrones de diseño | 40 |
| 3.4 Modelo de análisis. | 41 |
| 3.4.1 Diagramas de clases del análisis. | 41 |
| 3.4.2 Diagramas de Interacción (Colaboración). | 42 |
| 3.5 Modelo de diseño. | 44 |
| 3.5.1 Diagramas de clases del diseño con extensiones web. | 44 |
| 3.6 Diseño de la Base de Datos. | 46 |
| 3.6.1 Modelo lógico de datos. | 47 |
| 3.6.2 Descripción de las tablas de la base de datos. | 47 |
| 3.7 Diagrama de despliegue..... | 53 |
| 3.8 Conclusiones parciales. | 54 |
| CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBAS..... | 55 |
| 1.1 Introducción. | 55 |
| 1.2 Diagrama de componentes. | 55 |
| 1.3 Pruebas de software. | 57 |
| 1.3.1 Diseño de Casos de Prueba..... | 57 |

| | | |
|-----|--|-----------|
| 1.4 | Resultados Obtenidos..... | 60 |
| 1.5 | Conclusiones parciales..... | 61 |
| | CONCLUSIONES GENERALES..... | 62 |
| | RECOMENDACIONES..... | 63 |
| | Referencias Bibliográficas..... | 64 |
| | Bibliografía..... | 67 |
| | Anexos..... | 69 |

Índice de Figuras

| | |
|--|----|
| <i>Ilustración 1: Representación del Modelo de Dominio.</i> | 26 |
| <i>Ilustración 2: Representación del Modelo de Casos de Uso del Sistema.</i> | 32 |
| <i>Ilustración 3: Representación del patrón MVC.</i> | 40 |
| <i>Ilustración 4: Diagrama de clases del análisis: CU_ Gestionar Ejercicio</i> | 42 |
| <i>Ilustración 5: Diagrama de clases del análisis: CU_ Resolver Ejercicio</i> | 42 |
| <i>Ilustración 6: Diagrama de clases del análisis: CU_ Gestionar Ejercicios</i> | 43 |
| <i>Ilustración 7: Diagrama de clases del análisis: CU_ Resolver Ejercicio</i> | 44 |
| <i>Ilustración 8: Diagrama de clases del diseño: CU_ Gestionar Ejercicio</i> | 45 |
| <i>Ilustración 9: Diagrama de clases del diseño: CU_ Resolver Ejercicio</i> | 46 |
| <i>Ilustración 10: Modelo lógico de datos</i> | 47 |
| <i>Ilustración 11: Modelo de Despliegue</i> | 53 |
| <i>Ilustración 12: Diagrama de componentes: CU_ Gestionar ejercicio</i> | 56 |
| <i>Ilustración 13: Diagrama de componentes: CU_ Resolver Ejercicio</i> | 56 |
| <i>Ilustración 14: Diagrama de clases del análisis: CU_ Autenticar usuario</i> | 69 |
| <i>Ilustración 15: Diagrama de clases del análisis: CU_ Gestionar Profesores</i> | 69 |
| <i>Ilustración 16: Diagrama de clases del análisis: CU_ Gestionar Grupo</i> | 70 |
| <i>Ilustración 17: Diagrama de clases del análisis: CU_ Asignar Ejercicio</i> | 70 |
| <i>Ilustración 18: Diagrama de clases del análisis: CU_ Generar Reporte</i> | 71 |
| <i>Ilustración 19: Diagrama de clases del análisis: CU_ Autenticar Usuario</i> | 71 |
| <i>Ilustración 20: Diagrama de clases del análisis: CU_ Gestionar Profesores</i> | 72 |
| <i>Ilustración 21: Diagrama de clases del análisis: CU_ Gestionar Grupo</i> | 73 |
| <i>Ilustración 22: Diagrama de clases del análisis: CU_ Asignar Ejercicios.</i> | 73 |
| <i>Ilustración 23: Diagrama de clases del análisis: CU_ Generar Reporte</i> | 74 |
| <i>Ilustración 24: Diagrama de clases del diseño: CU_ Generar Reporte</i> | 74 |
| <i>Ilustración 25: Diagrama de clases del diseño: CU_ Gestionar Profesores</i> | 75 |
| <i>Ilustración 26: Diagrama de clases del diseño: CU_ Asignar Problema</i> | 75 |
| <i>Ilustración 27: Diagrama de clases del diseño: CU_ Autenticar Usuario</i> | 76 |
| <i>Ilustración 28: Diagrama de clases del diseño: CU_ Gestionar Grupo</i> | 76 |

INTRODUCCIÓN

Una de las herramientas más poderosas e indispensables en el desarrollo social y económico de cualquier país, vinculada de forma directa en la informatización y automatización de la mayoría de las esferas, procesos de la producción y los servicios en la actualidad, es el uso de las Tecnologías de la Información y las Comunicaciones (TIC). La educación no está ajena a este fenómeno actual ya que el empleo de estas aporta múltiples ventajas en la mejoría de la calidad docente, materializadas en aspectos tales como el acceso a la información desde áreas remotas, la flexibilidad en tiempo y espacio para el desarrollo de diferentes actividades. Su aplicación constituye un factor de motivación para los estudiantes, captando la atención de estos y convirtiéndose en uno de los motores del aprendizaje, ya que incita a la actividad y desarrollo del pensamiento.

Nuestro país ha experimentado un gran avance tecnológico en los últimos años, sobre todo en la esfera educacional, donde se han desarrollado sistemas que permiten un mejor entendimiento por parte de los estudiantes de varias temáticas impartidas. A pesar de esto, aún se sigue trabajando en el desarrollo de herramientas que ofrezcan la posibilidad de encontrar soluciones novedosas ante los desafíos sociales de la era moderna. El uso de las TIC en el proceso de enseñanza-aprendizaje en Cuba se ha incrementado pues las ventajas que proporciona en las distintas áreas y en particular en el estudio de la Física son muchas, tales como: mejora de la calidad de los contenidos, contar con laboratorios de forma virtual que son difíciles de tener, emplear otros métodos de autoevaluación para los mismos estudiantes y el poder contar con herramientas técnicas que apoyen la docencia.

La Universidad de las Ciencias Informáticas (UCI), cuenta con un nuevo modelo de formación centrado en el auto aprendizaje, el cual aboga por el uso de aplicaciones que permitan una mayor exploración de los contenidos por parte de los estudiantes.

La asignatura de Física, es hoy una de las que más problemas causan en los estudiantes, entre cosas, por su carácter abstracto en algunos contenidos, lo que requiere un especial esfuerzo por parte del profesorado para lograr la atención y la estimulación del aprendizaje de los alumnos. Este rechazo se debe a diferentes motivos, entre los que se encuentran, la falta de nivel en matemáticas; la enorme dificultad que conlleva para la mayoría seguir los razonamientos utilizados; y la tradicional forma de evaluación sin el uso de la tecnología.

Es por ello que la Universidad introduce la plataforma libre Moodle con el Entorno Virtual de Aprendizaje (EVA) como una de las principales aplicaciones destinadas a la motivación de la docencia.

Aunque en años anteriores se han desarrollado investigaciones relacionadas con esta problemática; actualmente en la Facultad 5 de la UCI, la asignatura de Física carece de aplicaciones informáticas propias, es decir, hechas en la Facultad, que le permita a los estudiantes observar los fenómenos físicos que suceden al aplicar la Ley de Gauss, contenido que es esencial para el entendimiento de la asignatura. Otro de los problemas que está influyendo en la enseñanza de esta disciplina es que los profesores tienen muy pocas formas de evaluar a los estudiantes en dicho contenido antes del examen parcial, debido a la estructura del Modelo de Planificación y Control del Proceso Docente (P1), el cual limita grandemente las horas de estudio dedicadas a dicho contenido. Además, los estudiantes al realizar ejercicios con diferentes niveles de complejidad, no conocen cuál o cuáles fueron los pasos en los que presentaron dificultades antes de arribar a la solución final del problema, ni el lugar al cual deben remitirse para solucionar este; por lo que si desean comprobar la solución, deben mostrárselo al profesor de manera presencial, lo que limita el avance de los estudiantes durante su auto preparación. Aunque existe el tiempo de atención diferenciada, este es insuficiente, atentando en la valoración de la situación real que pueda presentar cada estudiante. Otro factor importante que atenta contra la enseñanza es la no existencia de turnos de laboratorios que permitan demostrar los fenómenos que suceden en dicho contenido, donde el programa analítico de la asignatura en ese tema solo cuenta con una conferencia y una clase práctica.

Debido a la situación existente se plantea como **Problema Científico**: ¿Cómo mejorar el proceso de enseñanza aprendizaje de la asignatura de Física para los estudiantes de la Facultad 5?

Para dar solución al problema planteado se determina como **Objeto de estudio** de la investigación: Sistemas informáticos utilizados durante el proceso de enseñanza-aprendizaje en la asignatura de Física. Se define como **Objetivo general de la investigación**: Desarrollar una aplicación web para fortalecer la preparación individual de los estudiantes en el tema de Ley de Gauss dentro del proceso de enseñanza centrado en el aprendizaje.

Todo esto conlleva a determinar como **Campo de acción**: Los sistemas informáticos utilizados durante el proceso de enseñanza-aprendizaje en la asignatura de Física para el contenido de la Ley de Gauss.

Para guiar la investigación se define la siguiente **Idea a defender**: Con el desarrollo del sistema informático para la preparación individual de los estudiantes en el tema de la Ley de Gauss en la

asignatura Física en la Facultad 5, se logrará una mejor preparación individual de los estudiantes y un mejor proceso de enseñanza aprendizaje.

Para lograr el cumplimiento del objetivo general se proponen las siguientes **Tareas investigativas:**

1. Elaboración del marco teórico a partir del estado del arte actual referente al tema.
2. Selección de la metodología pedagógica para el desarrollo de la aplicación web.
3. Selección de la metodología, herramientas de software y lenguajes de programación para el desarrollo de la aplicación web.
4. Modelación del negocio y las actividades del flujo de trabajo de requerimientos.
5. Generación de los artefactos que corresponden a los flujos de trabajo modelo de negocio, requerimientos e implementación.
6. Implementación de la aplicación web.
7. Validación de los resultados.

Para llevar a cabo estas tareas se emplearon **métodos teóricos y empíricos, los cuales se detallan a continuación.**

Los **métodos teóricos** utilizados para cumplir con las tareas a desarrollar son:

- Método **Análisis histórico-lógico:** Este método es usado ya que permite constatar cómo ha evolucionado el desarrollo de las TIC vinculado al proceso de enseñanza-aprendizaje de la Física.
- Método **Análisis y síntesis:** Este método es utilizado con el objetivo de analizar la situación problemática como un todo, logrando dividir el objeto de estudio para una mejor comprensión, analizando los sistemas informáticos utilizados en la enseñanza aprendizaje en el campo de la Física, obteniendo la síntesis de los principales factores que intervienen en la disciplina.
- Método **Análisis documental:** Este método fue utilizado con el objetivo de seleccionar la documentación más acorde al contenido tratado en el sistema y de ahí lograr seleccionar los pasos y ejercicios más relevantes de los documentos expresados, a fin de expresar el tema implícito sin ambigüedades.

Los **métodos empíricos** utilizados para obtener información sobre el objeto de estudio fueron:

- **Observación:** Este método fue utilizado con el objetivo de conocer cómo se comportaba la utilización de los sistemas informáticos en el aprendizaje de la Física en la Facultad. Fue necesaria la observación para conocer las principales deficiencias que existían en los estudiantes de segundo año en el contenido de la Ley de Gauss.
- **Entrevista:** Este método fue utilizado con el objetivo de conocer por los estudiantes y profesores cómo les gustaría la inserción del empleo de las TIC y la aplicación propuesta para lograr el

mejoramiento de la asignatura de Física y una mayor motivación por la disciplina. El uso de la entrevista además permitió revelar las relaciones esenciales y las características fundamentales del objeto de estudio.

- **Encuesta:** Este método fue utilizado con el objetivo de tener una técnica de adquisición de información de interés sociológico, mediante un cuestionario previamente elaborado, a través del cual se pretende conocer el nivel de satisfacción de los estudiantes una vez que interactúa con la aplicación propuesta.

Estructura capitular

Capítulo 1: Fundamentación Teórica. Se exponen los elementos teóricos que sustentan el problema científico y los objetivos del trabajo. Se realiza un análisis de las metodologías y herramientas de desarrollo que se pueden utilizar y se justifica la selección de cada una de ellas.

Capítulo 2: Características del Sistema. Se especifican los requisitos funcionales y no funcionales que debe cumplir el sistema así como el diagrama de casos de uso y las descripciones de ellos.

Capítulo 3: Análisis y Diseño del Sistema. Se realiza el análisis y diseño del sistema donde se muestran los diagramas de clases del análisis, así como los diagramas de clases del diseño, expresados en diagramas de clases con estereotipos web, además del modelo de datos lógicos y físicos para la construcción de la Base de datos.

Capítulo 4: Implementación y Pruebas. Los aspectos del desarrollo son tratados en este capítulo. Se describe cómo está implementado el sistema. Además, se definen los tipos de pruebas y los casos de prueba que se le realizarán al software.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA**1.1 Introducción.**

En este capítulo se realizará el estudio del arte correspondiente a diversos temas como: el proceso de enseñanza aprendizaje y la incorporación de las TIC en la educación sobre todo en el campo de la Física, han influido en lograr un sistema educacional centrado en el uso de las tecnologías, las definiciones y comparaciones entre tareas, problemas y ejercicios, todo un estudio de las diferentes aplicaciones existentes en el mundo enfocadas de una forma u otra con el tema de la investigación. Además, se describirá un conjunto de tecnologías, metodologías, herramientas, lenguajes de programación y Sistemas Gestores de Base de Datos (SGBD) más utilizados, para definir y seleccionar los que se utilizarán en el desarrollo de este trabajo de diploma.

1.2 Proceso de enseñanza aprendizaje (PEA).

En nuestra sociedad cada vez se hace más evidente la necesidad de encontrar vías que permitan a los estudiantes de distintos niveles, asimilar los sistemas de conocimientos y los métodos de la actividad intelectual y práctica, que los pongan en posición de dar respuestas a situaciones que se les presenten en la vida con perseverancia y afán por lograr el objetivo.

El proceso de enseñanza aprendizaje (PEA), requiere de una planificación y dirección por parte del profesor, conducción en la que están presentes tres elementos fundamentales: la orientación, la ejecución y el control.

El propósito esencial de la enseñanza aprendizaje es la transmisión de información mediante la comunicación directa o soportada en medios auxiliares, que presentan un mayor o menor grado de complejidad y costo.

Este proceso ha sido históricamente caracterizado de formas diferentes, que van desde la identificación como proceso de enseñanza con un marcado énfasis en el papel central del profesor como transmisor de conocimientos, hasta las concepciones más actuales en la que se concibe el proceso de enseñanza aprendizaje como un todo integrado en el que se pone de relieve el papel protagónico del educando.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Dicho proceso es una unidad dialéctica entre la instrucción y la educación; igual característica existe entre el enseñar y el aprender. Todo el proceso de enseñanza aprendizaje tiene una estructura y un funcionamiento sistémicos, es decir, está conformado por elementos o componentes estrechamente interrelacionados. Este enfoque conlleva a realizar un análisis de los distintos tipos de relaciones que operan en mayor o menor medida en los componentes del proceso antes mencionado.

1.3 Proceso de enseñanza aprendizaje de la Física en el nivel superior.

El proceso de enseñanza aprendizaje de la Física en cualquier nivel educativo y en especial en el universitario no ha sido estático, sino una constante conformación dialéctica de sus componentes), de acuerdo a la época e ideología vigente con el fin único de acercar esta ciencia al educando, determinando que: *“...la educación superior tienen hoy el encargo social de desarrollar individuos capaces de prepararse toda la vida, de manera autodidacta, para resolver los problemas de la época apoyados en una sólida formación básica. Y no es posible cumplir esta tarea sin transformar profundamente la metodología del proceso de enseñanza - aprendizaje.”*(1)

En la actualidad, las principales investigaciones sobre el PEA de la Física convergen en la idea de que el mismo se organice atendiendo a la relación objetiva entre cultura, educación y ciencia. En este caso, formar un individuo identificado por el ambiente científico tecnológico y social que caracteriza a la sociedad contemporánea, donde las formas de pensar y hacer, deben estar dirigidas a un ser proactivo en cuanto a la gestión del conocimiento y a la solución de problemas.

Breve reseña sobre el proceso de enseñanza aprendizaje de la Física en Cuba y en la UCI.

Desde mediados del siglo XX y hasta la actualidad, el PEA de la Física en Cuba se encuentra influenciado básicamente, por la escuela soviética (aunque la influencia "Occidental" no ha sido despreciable), caracterizándose por una sólida formación básica teórica y experimental, incluyendo la investigación científica desde los primeros años de las carreras universitarias.

Desde el propio triunfo de la Revolución Cubana, se evidenció la necesidad de recuperar la formación científica de las nuevas generaciones desde una universidad la cual, por factores políticos y sociales diversos, presentaba un considerable atraso como se hace constar en un memorando al rector de la Universidad de La Habana, citado por Baracca, A y otros *“... toda la nueva Escuela de Física cabía en la maleta del Director, las condiciones iniciales eran pésimas. No había ni equipos de laboratorio, ni profesores, ni alumnos bien preparados, ni instalaciones e inmuebles adecuados.”*(2)

Esta ingente necesidad provoca la experimentación por parte del sistema educativo cubano, de las distintas corrientes o tendencias en el PEA de las ciencias fundamentalmente occidentales,

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

enriqueciendo las mismas de acuerdo a la propia experiencia cubana. Desde el primer momento, estas experiencias estuvieron dirigidas a la formación de un profesional comprometido con el proceso social y altamente calificado para solucionar problemáticas de cualquier índole.

En la actualidad, en Cuba se puede apreciar una orientación hacia la tendencia denominada: Enfoque Sociocultural del Proceso de Enseñanza Aprendizaje de las Ciencias.

La UCI, desde su propio surgimiento en el curso 2002-2003, fue adoptando de forma paulatina la tendencia constructivista para la enseñanza de la Física. Esta estuvo caracterizada desde el inicio por la presencia de un sitio web mediante el cual se desarrollaba la docencia a través de simulaciones informáticas y un conjunto de situaciones problemáticas, las cuales tenían que ser solucionadas por los estudiantes actuando el profesor como mediador.

Posteriormente, se introduce el programa vigente en la CUJAE, conformado por dos asignaturas: Física I y II teniendo como medio principal el Entorno Virtual de Enseñanza y Aprendizaje (EVE/A).

A partir del curso 2009-2010 se reorienta el PEA hacia un modelo de formación centrado en el aprendizaje con tres principios básicos de la teoría pedagógica cubana: la vinculación del estudio con el trabajo, el vínculo entre la instrucción y la educación y la adopción del método de las ciencias como vía fundamental para la solución de problemas, todo ello mediado por el entorno productivo y tecnológico.

Se labora para que las tareas de aprendizaje propicien el desarrollo potencial de los estilos de aprendizaje para garantizar así el carácter más personalizado de las actividades de enseñanza aprendizaje y que la evaluación y el control dentro del PEA presenten un enfoque integral que se ejecute de forma integrada con los procesos de producción e investigación, combinando el conocimiento, la comprensión reflexiva, la solución de problemas, las habilidades técnicas, las actitudes y la ética de la evaluación. Una evaluación integrada y holística caracterizada por estar orientada a problemas, ser interdisciplinaria, cubrir grupos de competencias, exigir habilidades analíticas y combinar la teoría con la práctica.

Estas características de los componentes del PEA de la Física en la UCI, permiten una adecuada estructuración del proceso formativo, sin embargo, las mismas no se ponen en práctica con el rigor que se requiere y no está exenta de dificultades en la orientación, seguimiento y control del trabajo independiente de los estudiantes; en el aprovechamiento de las potencialidades que ofrece el EVE/A como complemento para la auto preparación de profesores y estudiantes.

Como se aprecia, la enseñanza y aprendizaje de las ciencias y en especial de la asignatura Física, mantienen una vitalidad profunda en lo que respecta a la necesidad imperiosa de formar un individuo

acorde a las condiciones del mundo actual donde la ciencia, la tecnología y los problemas sociales y políticos predeterminan el mismo.

1.4 La utilización de las TIC en la educación.

La Educación Superior en el mundo y en Cuba, en estrecho vínculo con las necesidades de la sociedad cambiante, ha utilizado las potencialidades de las Tecnologías de la Información y las Comunicaciones.

Nuestro país ha puesto en práctica disímiles estrategias para la implementación y asimilación de las TIC en el sistema educacional, diferenciando los distintos niveles de enseñanza, todas a partir de transformaciones realizadas en sus respectivas infraestructuras, así como en los currículos de cada centro o institución.

A partir de ese instante se trazaron líneas directrices del proyecto de Informatización de la educación que permitieron la elaboración de los nuevos planes y programas de estudio de Informática en los diferentes subsistemas de la educación cubana. La primera experiencia se llevó a cabo en la Universidad de Camagüey (1996), donde se desarrolló un curso a distancia a través del correo electrónico. En ese momento fue posible constatar nuevas necesidades que marcarían un punto vital en la incorporación de las TIC a la educación y que por su importancia debían ser producto de transformaciones, entre ellas se tiene:

- Nuevas concepciones en el diseño de materiales didácticos.
- Desarrollo de estrategias de aprendizaje en los nuevos entornos y nuevas formas de comunicación.
- Cambios del rol del profesor y la implicación en sus competencias profesionales.

Con el fortalecimiento de la infraestructura Informática en cada una de las universidades fue necesario que un grupo de profesores, con experiencia en la educación superior, se enfrascara en la búsqueda de nuevos modelos y metodologías que permitieran la inserción de estos avances de la tecnología en el ámbito educacional.

En la bibliografía consultada se aprecian evidentes ventajas que propician las TIC a nivel universitario y que muchos autores como Leyet, O. (2002), Castañeda, E. (2003), Rodríguez, J. (2003), Castañeda, E. (2006), Moreira, M. (2008), Arcia, W. (2006) y Amaya, D., (2008) entre otros, han destacado que:

1. Todas las tendencias o corrientes pedagógicas pueden hacer uso de las tecnologías en diferentes formas, y los resultados de su introducción en el proceso de enseñanza-aprendizaje

pueden servir incluso para potenciar las corrientes más perjudiciales, tradicionales, o de cualquier otro tipo que pueda pensarse, propiciando posiblemente una mayor efectividad de las mismas, cualesquiera sean estas.

2. Ninguna herramienta tecnológica usada en el proceso de enseñanza aprendizaje por sí misma conlleva a mejoras en el aprendizaje, lo que las convierte en herramientas útiles o no en el proceso de aprendizaje es cómo se usan en contextos sociales específicos a partir del diseño instruccional en que se insertan.
3. Permite el acercamiento de los alumnos a los contenidos de forma menos rígida, favoreciendo su rápida familiarización con las máquinas y el uso de software, aumenta su autonomía, tienen acceso a más información.
4. El alumno se siente más participe de su aprendizaje, comprende la necesidad de su constante búsqueda de información y análisis de la misma, logrando así establecer su propio ritmo de aprendizaje.
5. El profesor cuenta con herramientas técnicas más avanzadas que facilitan la exposición.
6. Se mejora la comunicación entre los integrantes del PEA. La misma se establece cuando sea necesario, a través de correo electrónico, chat, o foro de discusión entre el profesor y el alumno, o entre alumno- alumno”.

Es necesario destacar que el uso de las TIC permite una educación continua durante toda la vida y ofrece vías para la superación y formación profesional y cultural, favoreciendo el empleo de la educación a distancia.

1.5 Definición de tareas, problemas y ejercicios.

¿Cómo resolver problemas?, es una de las incógnitas que ha sido estudiada por numerosos científicos en el área de la educación, tanto del extranjero, como de nuestro país. A esta tarea se han dedicado psicólogos, pedagogos y personal ocupado de las didácticas particulares. Cuando se revisa la literatura psicológica y pedagógica sobre el tema, aparecen con mucha frecuencia los conceptos de tarea, problema y ejercicio. Unas veces los autores definen algunos de ellos, pero en ocasiones no se dejan claras sus características comunes y sus diferencias. Pero lo que sí ocurre es que, casi siempre, los profesores no diferencian entre unos y otros y utilizan estos términos incorrectamente. No faltan los casos de docentes que no pueden decir correctamente en qué consiste cada uno de ellos, y por eso se escucha hablar de problemas en casos que se trata de simples ejercicios y viceversa. Por tanto, se comenzará haciendo una precisión de las similitudes y diferencias, que

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

existen entre estos conceptos.

Por tarea docente entendemos cualquier trabajo o actividad cognitiva que deba resolver el alumno, orientada por el profesor o que se plantee solucionar él mismo, de acuerdo con sus intereses y motivaciones. Según el grado de complejidad de las tareas, puesto de manifiesto en la participación más o menos activa que tenga que hacer el alumno de su pensamiento para resolverlas, pueden ser clasificadas como problemas o ejercicios. (2)

Actualmente existen varias definiciones de problema de acuerdo con los elementos que le interesa resaltar a cada autor. La definición a utilizar en esta investigación sobre lo que constituye un problema docente se realiza a continuación: es una tarea que se le plantea al alumno dándosele determinadas condiciones y datos (o sin estos últimos), en la que él debe buscar una respuesta o resultado (incógnita), que puede ser cuantitativo o cualitativo, y cuyo método o vía de solución es desconocido por él, pero que, de forma general, posee el sistema de conocimientos y habilidades que le permiten enfrentar dicha búsqueda. (2)

Mientras que por ejercicio, se consideran aquellas tareas cuyos métodos y vías de solución son conocidos por el alumno y que persiguen como objetivo, además de fijar los conocimientos, contribuir a la automatización de determinadas acciones y algoritmos que le son necesarios para el cumplimiento rápido y eficaz de determinadas actividades teóricas y prácticas. (2)

Las diferencias fundamentales entre ejercicios y problemas, según la Dra. María Luz Callejo de la Vega, se mencionan en la siguiente tabla:(2)

| Ejercicios | Problemas |
|---|---|
| 1. De un golpe de vista se ve inmediatamente o casi inmediatamente en qué consiste la cuestión y cuál es el medio para resolverla. | 1. A primera vista no se sabe cómo atacar el problema y resolverlo; a veces incluso no se ve claro en qué consiste el problema. |
| 2. El objetivo principal del ejercicio es aplicar de forma rutinaria conocimientos y mecanismos ya conocidos y fáciles de identificar. | 2. Para resolver el problema no basta con aplicar una regla o una “receta” de forma rutinaria, sino que a fuerza de búsqueda y de intuición hay que elaborar una solución profundizando en el conjunto de conocimientos y experiencias anteriores. |
| 3. En general, la resolución de un ejercicio exige poco tiempo. | 3. En general, la resolución de un problema exige más tiempo. |

| | |
|--|---|
| 4. No se suelen considerar los aspectos afectivos durante la resolución de un ejercicio: motivación por resolverlo, interés en su solución, etc. | 4. La resolución de un problema exige una inversión importante de energía y de afectividad: frustración inicial, voluntad de resolverlo, perseverancia en su solución, etc. |
| 5. En general son tareas cerradas. | 5. El problema puede ser más o menos abierto o cerrado. |
| 6. Los ejercicios abundan en los libros de texto. | 6. Los problemas son escasos en los libros de texto. |

Desde el punto de vista de la función didáctica, la diferencia entre los ejercicios y los problemas es que los primeros se utilizan para desarrollar habilidades y mecanizar procedimientos, mientras que los segundos son utilizados, de forma general, para contribuir al desarrollo del pensamiento de los alumnos.

1.6 Sistemas informáticos de apoyo al aprendizaje de la Física existentes en Cuba y el mundo.

1.6.1 EDUFI en Energía.

Es un producto informático cubano para la Enseñanza de la Física en Energía, es un sitio WEB que recoge contenidos y ejercicios de energía en las diferentes educaciones, artículos y bibliografías adicionales relacionados con el tema, además se acompañan de imágenes y videos así como un glosario que propician la comprensión de los términos principales. En su diseño se empleó el sistema de gestión de contenidos Joomla. El cual cuenta con numerosas ventajas dentro de las que se encuentran el poder instalarse sobre cualquier plataforma de sistema operativo (multiplataforma), permite editar el contenido de un sitio Web de manera sencilla. Dicha aplicación de código abierto programada mayoritariamente en PHP bajo una licencia GPL. Este administrador de contenidos puede trabajar en Internet o intranets y requiere de una base de datos MySQL, así como, preferiblemente, de un servidor HTTP Apache.(3)

1.6.2 C@mpus Virtual.

Este proyecto intenta acercar la tecnología al ámbito docente, entre ellos la Física, es capaz de funcionar en un PC enlazada en una red local. Es perfecto para administrar grupos de unos 50 alumnos. Esta herramienta posibilita al profesor llevar el control del alumnado, así como la publicación de noticias, actividades, material de clase, bibliografía, direcciones de interés, creación de debates entre los alumnos, además de permitir la realización de tutorías, totalmente integradas en

la aplicación, e incluso se pueden proponer exámenes para que el alumno sea capaz de autoevaluarse y controlar cómo lo está haciendo. Esta herramienta se encuentra permanentemente en desarrollo. (4)

1.6.3 Gauss View.

Gauss View es la interfaz gráfica más avanzada y poderosa disponible para Gauss.

Con Gauss View, puede configurar, poner en marcha, seguimiento y control de los cálculos de Gauss, recuperar y ver los resultados, todo sin salir de la aplicación. Gauss View 5 incluye muchas nuevas características diseñadas para facilitar el trabajo con los grandes sistemas de interés físico y químico.

También proporciona soporte completo para todos los nuevos métodos de modelado y funciones de Gauss. Proporciona características para todas las fases de estudio de gran tamaño molecular de sistemas, la importación de las moléculas de los archivos de AP, a través de la modificación de las características estructurales y la creación de cálculos ONIOM de Gauss 09, de visualización y trazado de los resultados finales. Gauss View También puede importar muchos otros formatos populares de cambio de la estructura. (5)

1.6.4 Gauss 09W.

Gauss 09W es la de Gauss 09 versión para ordenadores Windows. Gauss 09 es la versión más reciente de la serie de Gauss ®.

Principales Características:

- Investigaciones integrales de las moléculas y de sus reacciones
- De diseño complejo, pero fácil de usar, características: la automatización, presentación intuitiva de los resultados. (5)

1.6.5 Entornos Virtuales de Aprendizaje.

El desarrollo de aplicaciones informáticas en las universidades cubanas ha servido de gran apoyo a la docencia. Algunas de ellas cuentan con un Moodle también conocido como Entorno Virtual de Aprendizaje (EVA) dentro de las que se encuentran: La Universidad de La Habana, la Universidad Central de Las Villas y la Universidad de Holguín. Estos entornos virtuales ofrecen al estudiante materiales donde pueden obtener conferencias, bibliografías de apoyo, libros, entre otros, se pueden encontrar ejercicios para resolver, así como pruebas parciales que los estudiantes utilizan para su auto preparación y también cuentan con los llamados foros donde pueden abordar temas de las

asignaturas. En la Universidad de las Ciencias Informáticas (UCI) también se cuenta con un EVA que ofrece las mismas posibilidades y posee las características muy parecidas a los entrenadores de las universidades antes mencionadas, brindando para la asignatura de Física una estructura que les da la posibilidad a los estudiantes de mantenerse informados sobre los próximos eventos a realizar, además de mantener un control de forma general del trabajo que desarrollan los educandos.

1.7 Metodologías, herramientas y lenguajes de modelado a utilizar.

1.7.1 Metodología de desarrollo de software.

Todo proceso de desarrollo de una aplicación tiene como propósito la producción eficaz y eficiente de un producto software que reúna los requisitos determinados por el cliente. Un proceso de software detallado y completo suele denominarse *Metodología*. (6)

Las metodologías de desarrollo de software se han clasificado en dos grandes grupos: las ágiles y las tradicionales. Las metodologías ágiles enfatizan la comunicación con el cliente mientras que suelen ser criticadas por la falta de documentación técnica. Las tradicionales o pesadas son aquellas con mayor énfasis en la planificación y control del proyecto.

Entre las metodologías ágiles se destacan XP (*eXtreme Programming*), OpenUP (*Open Unified Process*), Scrum y Crystal. Así mismo se puede mencionar RUP (*Rational Unified Process*) y MSF (Microsoft Solution Framework) como las más destacadas dentro de las metodologías pesadas.

A continuación se realiza un análisis de algunas metodologías, para de ese modo definir cuál es la indicada para el utilizar en el desarrollo del sistema que se quiere lograr.

1.7.1.1 Extreme Programming (XP).

XP consiste en una programación rápida o extrema, cuyo rasgo característico es tener como parte del equipo de desarrollo al usuario final (cliente), pues es uno de los requisitos para llegar al éxito del proyecto. Creada por Kent Beck, impulsor del manifiesto ágil.

Esta metodología se basa en:

- Pruebas Unitarias: Pruebas realizadas a los principales procesos.
- Refabricación: Reutilización de código. Son creados patrones o modelos estándares que traen consigo una mayor flexibilidad al cambio.
- Programación en pares: Dos desarrolladores participan en un proyecto en una misma estación de trabajo. Cada miembro lleva a cabo la acción que el otro no está haciendo en ese momento.

El ciclo de desarrollo requiere los siguientes pasos:

- El cliente define el valor de negocio a implementar.
- El programador estima el esfuerzo necesario para su implementación.
- El cliente selecciona qué construir, de acuerdo a sus prioridades y las restricciones de tiempo.
- El programador construye ese valor de negocio.
- Vuelve al paso 1.

El ciclo de vida ideal de XP consiste de seis fases: Exploración, Planificación de la Entrega (*Release*), Iteraciones, Producción, Mantenimiento y Muerte del Proyecto. (6)

1.7.1.2 Rational Unified Process (RUP).

Rational Unified Process (RUP) apareció en 1998, creado por James Rumbaugh, Grady Booch e Ivar Jacobson para la *Rational Corporation*. El Proceso Unificado de Rational es un proceso de desarrollo de software. El Proceso Unificado es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto.

El Lenguaje Unificado de Modelado (Unified Modeling Language, UML) es utilizado para modelar todos los diagramas de un sistema software.

Las principales características con que cuenta el Proceso Unificado de Desarrollo de Software son: dirigido por casos de uso, centrado en la arquitectura e iterativo e incremental. Los mismos son descritos a continuación:

- Dirigido por casos de uso: El proceso de desarrollo está dirigido por casos de uso, sigue un hilo que avanza a través de una serie de flujos de trabajo que parten de los casos de uso.
- Centrado en la arquitectura: La arquitectura es el elemento central a partir del cual el sistema se desarrolla y evoluciona, no solo en su desarrollo inicial, sino también a lo largo de las futuras iteraciones y generaciones, por lo que esta debe permitir el correcto desarrollo de todos los casos de uso requeridos tanto en el presente como en el futuro.
- Iterativo e incremental: En RUP el trabajo se divide en partes más pequeñas o mini-proyectos llamados iteraciones, con la culminación de cada iteración se logra un incremento o crecimiento del producto.

RUP se divide en 4 fases:

- Inicio: El objetivo de esta etapa es determinar la visión del proyecto.
- Elaboración: El objetivo es determinar la arquitectura óptima.
- Construcción: El objetivo es obtener la capacidad operacional inicial.

- Transición: El objetivo es obtener la primera versión del proyecto (Release).

Flujos de Trabajo de Ingeniería:

- Modelamiento del negocio: Entendiendo las necesidades del negocio.
- Requerimientos: Traslado de las necesidades del negocio a un sistema automatizado.
- Análisis y Diseño: Traslado de los requerimientos dentro de la arquitectura de software.
- Implementación: Creando software que se ajuste a la arquitectura y que tenga el comportamiento deseado.
- Pruebas: Asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado está presente.
- Despliegue: Hacer todo lo necesario para la salida del proyecto.

Flujos de Trabajo de Soporte:

- Administración de configuración y cambios: Guardando todas las versiones del proyecto.
- Administración de proyectos: Administrando horarios y recursos.
- Ambiente: Administrando el ambiente de desarrollo. (6)

1.8 Lenguajes de modelado:

1.8.1 Lenguaje Unificado de Modelado (UML).

El Lenguaje Unificado de Modelado conocido como UML (por sus siglas en inglés Unified Modeling Language), es un lenguaje utilizado para visualizar, especificar, construir y documentar los artefactos de un sistema. Es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad. Permite representar en mayor o menor medida todas las fases de un proyecto informático: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos, hasta la implementación y configuración en los diagramas de despliegue. (7)

Características de UML:

- Modela estructuras complejas.
- Lenguaje distribuido y adecuado a las necesidades de conectividad actual y futura.
- Reemplaza a decenas de notaciones empleadas con otros lenguajes.
- Las estructuras más importantes que soporta tienen su fundamento en las tecnologías orientadas a objetos, tales como objetos, clase, componentes y nodos.
- Comportamiento del sistema: casos de uso, diagramas de secuencia y de colaboraciones.

Es importante destacar que un modelo UML modela gráficamente lo que hará el sistema, pero no dice cómo llevar a cabo la implementación del mismo. (8)

1.8.2 Business Process Modeling Notation (BPMN).

Business Process Modeling Notation (BPMN) es una notación gráfica que describe la lógica de los pasos de un proceso de negocio. Esta notación ha sido especialmente diseñada para coordinar la secuencia de los procesos y los mensajes que fluyen entre los participantes de las diferentes actividades, proporcionando un lenguaje común para que las partes involucradas puedan comunicar los procesos de forma clara, completa y eficiente. De esta forma BPMN define la notación y semántica de un Diagrama de Procesos de Negocio (Business Process Diagram, BPD). Permite además modelar los procesos de una manera unificada y estandarizada permitiendo un entendimiento a todas las personas de una organización. (9)

1.9 Herramientas CASE.

Las herramientas CASE (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el coste de las mismas en términos de tiempo y de dinero. Estas herramientas nos pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costes, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras. (10)

Estas herramientas pueden proveer muchos beneficios en todas las etapas del proceso de desarrollo de software, algunos de ellos son:

1. Verificar el uso de todos los elementos en el sistema diseñado.
2. Automatizar el diseño de diagramas.
3. Ayudar en la documentación del sistema.
4. Ayudar en la creación de relaciones en la Base de Datos.
5. Generar estructuras de código.

1.9.1 Visual Paradigm.

Es escogida esta herramienta para la modelación del sistema por las facilidades que brinda para el diseño UML del ciclo de vida completo del desarrollo de software: análisis, diseño, construcción, pruebas y despliegue, todo ello orientado a objetos.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

El software de modelado UML ayuda a una más rápida construcción de aplicaciones de calidad, mejores y a un menor coste. Posibilita el modelado de base de datos, requerimientos, proceso de negocio, permite realizar todo tipo de diagramas de clases, código inverso, generar códigos desde diagramas y generar documentación, entre otras funcionalidades. Visual Paradigm permite realizar, mediante un análisis textual, técnicas para la captura y análisis de requisitos, identificando casos de uso, actores y clases.

La versión a utilizar es Visual Paradigmfor UML 6.0 Enterprise Edition, la misma constituye una herramienta multiplataforma y permite modelar todos los artefactos que se obtendrán a partir del análisis del negocio y el sistema, además brinda numerosas funcionalidades como: (11)

Captura, especificación y análisis de requisitos mediante la creación de Modelos de Análisis Textual.

1. A partir de las descripciones contenidas en los análisis textuales se pueden identificar los elementos candidatos a incluir en los diagramas de casos de uso.
2. Es una herramienta colaborativa, es decir, soporta múltiples usuarios trabajando sobre el mismo proyecto.
3. Diseño centrado en casos de uso y enfocado al negocio que generan un software de mayor calidad.
4. Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
5. Permite control de versiones.
6. Genera la documentación del proyecto automáticamente en varios formatos como Web o PDF.

Para el desarrollo de la solución propuesta se ha seleccionado Visual Paradigm por las características anteriormente expuestas, además por la posibilidad de trabajar en la plataforma GNU/Linux. No presenta ningún problema con su licencia para la elaboración de software comercial, el cual es un aspecto de suma importancia ya que se desea un software desarrollado completamente con herramientas libres.

1.9.2 Rational Rose Enterprise.

Rational Rose Enterprise es el producto más completo de la familia Rational Rose. Todos los productos Rational Rose incluyen soporte Unified Modeling Language™ (UML™).

Rational Rose Enterprise es la mejor elección para el ambiente de modelado que soporte la generación de código a partir de modelos en Ada, C++, Visual C++ y Visual Basic. Como todos los demás productos Rational Rose, proporciona un lenguaje común de modelado para el equipo que facilita la creación de software de calidad más rápidamente. (12)

Características adicionales incluidas:

- Característica de control por separado de componentes modelo que permite una administración más granular y el uso de modelos
- La generación de código Ada, ANSI C ++, C++, CORBA, Java y Visual Basic, con capacidad de sincronización modelo- código configurables.
- Capacidad de análisis de calidad de código.
- El Add-In para modelado Web provee visualización, modelado y las herramientas para desarrollar aplicaciones de Web.
- Modelado UML para trabajar en diseños de base de datos, con capacidad de representar la integración de los datos y los requerimientos de aplicación a través de diseños lógicos y físicos.
- Publicación web y generación de informes para optimizar la comunicación dentro del equipo. (12)

1.10 Sistema de Gestión de Bases de Datos (SGBD).

Los sistemas de gestión de bases de datos (en inglés *database management system*, abreviado *DBMS*) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.

El propósito general de los sistemas de gestión de bases de datos es el de manejar de manera clara, sencilla y ordenada un conjunto de datos que posteriormente se convertirán en información relevante para una organización. (13)

1.10.1 PostgreSQL 8.3

Se ha seleccionado Sistema de Gestión de Base de datos a PostgreSQL 8.3ya que es un servidor de bases de datos relacional libre, ofrecido bajo la licencia BSD (*Berkeley Software Distribution*). PostgreSQL incluye características de la orientación a objetos, como puede ser: la herencia, tipos de datos, funciones, restricciones, disparadores, reglas e integridad transaccional. Las principales características de PostgreSQL son: (13)

- Atomicidad: Asegura que la operación se ha realizado o no, y por lo tanto ante un fallo del sistema no puede quedar a medias.
- Consistencia: Asegura que sólo se empiece aquello que se puede acabar. Se ejecutan las operaciones que no van a romper la reglas y directrices de integridad de la base de datos.
- Aislamiento: Asegura que una operación no puede afectar a otras. Esto asegura que dos transacciones sobre la misma información nunca generará ningún tipo de error.

- Durabilidad: Asegura que una vez realizada la operación, esta persistirá y no se podrá deshacer.
- Soporte de casi todos los sistemas operativos: Linux, Unix, Mac OS, Beos, Windows, etc.
- Documentación muy bien organizada, pública y libre.
- Altamente adaptable a las necesidades del cliente
- Soporte nativo para los lenguajes más populares del medio: PHP, C, C++, Perl, Python.
- Soporte de todas las características de una base de datos profesional.

1.10.2 MySQL.

MySQL es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C. Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.(14)

1.11 Servidor Web.

Un servidor web es un programa que sirve para atender y responder a las diferentes peticiones de los navegadores, proporcionando los recursos que soliciten usando el protocolo HTTP o el protocolo HTTPS.

Apache 2.0 es un servidor Web potente, flexible y disponible para distintas plataformas y entornos. Es altamente configurable de diseño modular, posibilitando que los administradores de sitios Web puedan elegir los módulos que serán incluidos y ejecutados en el servidor. (15)

Características de Apache:

- Es una tecnología gratuita, de código abierto y multi-plataforma, lo que proporciona transparencia en todo el proceso de instalación.
- Es prácticamente universal, por su disponibilidad en multitud de sistemas operativos.
- Facilita la integración como "plug-ins" de lenguajes de programación de páginas web dinámicas
- Posee una alta configurabilidad en la creación y gestión de logs, de este modo es posible tener un mayor control sobre lo que sucede en el servidor.

- Este servidor Web tiene una fácil integración con varios lenguajes de programación como: Java, Perl y especialmente PHP. Dicha relación a dado a lugar el desarrollo de aplicaciones como el APPSERV y XAMPP los cuales instalan el Apache y el PHP configurados para su uso. (15)

1.12 Lenguajes de programación para aplicaciones web.

Un lenguaje de programación es un idioma diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Definido además como un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Es utilizado para controlar el comportamiento físico y lógico de una computadora. (16)

1.12.1 Lenguajes de programación del lado del cliente.

Los lenguajes del lado del cliente son aquellos que pueden ser directamente comprendidos por el navegador y no necesitan un pre-tratamiento, son totalmente independientes del servidor, lo cual permite que la página pueda ser albergada en cualquier sitio. Dentro de los lenguajes del lado del cliente se encuentran el JavaScript (JScript), HTML, CSS, entre otros.

1.12.1.1 HTML.

HTML, siglas de *Hyper Text Markup Language* («lenguaje de marcado de hipertexto»), es el lenguaje de marcado predominante para la elaboración de páginas web, utilizado normalmente en la www (World Wide Web). Fue creado en 1986 por el físico nuclear Tim Berners-Lee. HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<,>). El entorno para trabajar HTML es simplemente un procesador de texto, como el que ofrecen los sistemas operativos Windows (Bloc de notas), o el que ofrece MS Office (Word).

El conjunto de etiquetas que se creen, se deben guardar con la extensión .htm o.html. Estos documentos pueden ser mostrados por los visores o "browsers" de páginas Web en Internet, como Netscape browser, Mozilla Firefox, Mosaic, Opera y Microsoft Internet Explorer. También existe el HTML Dinámico (DHTML), que es una mejora de Microsoft de la versión 4.0 de HTML que le permite crear efectos especiales en cada una de las páginas creadas. (17)

1.12.1.2 JavaScript.

JavaScript es un lenguaje de programación que permite a los desarrolladores crear acciones en sus páginas web. No requiere de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos. Entre sus principales características

podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros.

Nació con la necesidad de permitir a los autores de sitio web crear páginas que permitan intercambiar con los usuarios, ya que se necesitaba crear webs de mayor complejidad. El HTML solo permitía crear páginas estáticas donde se podía mostrar textos con estilos, pero se necesitaba interactuar con los usuarios. El código JavaScript podemos encontrarlo dentro de las etiquetas `<body></body>` de nuestras páginas web. Por lo general se insertan entre: `<script></script>`. También pueden estar ubicados en ficheros externos usando:

```
<script type="text/javascript" src="micodigo.js"></script>
```

JavaScript es soportado por la mayoría de los navegadores como Internet Explorer, Netscape, Opera, Mozilla Firefox, entre otros. (18)

1.12.1.3 CSS.

El nombre hojas de estilo en cascada viene del inglés *Cascading Style Sheets*, del que toma sus siglas. CSS es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). La idea que se encuentra detrás del desarrollo de CSS es separar la *estructura* de un documento de su *presentación*. Una hoja de estilos CSS consiste en una serie de *reglas*. Cada regla consiste en uno o más *selectores* y un *bloque de estilos* con los estilos a aplicar para los elementos del documento que cumplan con el selector que les precede. Cada bloque de estilos se define entre llaves, y está formado por una o varias declaraciones de estilo con el formato propiedad: valor. (17)

Algunas ventajas de utilizar CSS son:

- Control centralizado de la presentación de un sitio web completo con lo que se agiliza de forma considerable la actualización del mismo.
- Separación del contenido de la presentación, lo que facilita al creador, diseñador, usuario o dispositivo electrónico que muestre la página, la modificación de la visualización del documento sin alterar el contenido del mismo, sólo modificando algunos parámetros del CSS.
- Optimización del ancho de banda de la conexión, pues pueden definirse los mismos estilos para muchos elementos con un sólo selector; o porque un mismo archivo CSS puede servir para una multitud de documentos.(17)

1.13 Lenguaje programación del lado del servidor.

La programación del lado del servidor es una tecnología que consiste en el procesamiento de una petición de un usuario mediante la interpretación de un script en el servidor web para generar páginas HTML dinámicamente como respuesta. Entre los lenguajes de lado del servidor se encuentran aquellos que son reconocidos, ejecutados e interpretados por el propio servidor y que se envían al cliente en un formato comprensible para él, por ejemplo: ASP, PHP, JSP.(19)

1.13.1 PHP.

PHP es un acrónimo recursivo que significa *PHP Hypertext Pre-processor*. Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre. Es un lenguaje interpretado de propósito general ampliamente usado y que está diseñado especialmente para desarrollo web y puede ser incrustado dentro de código HTML. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno.

Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Mediante extensiones es también posible la generación de archivos PDF, Flash, así como imágenes en diferentes formatos. Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.(16)

1.14 Entorno de desarrollo (NetBeans IDE 6.9).

NetBeans es una plataforma para el desarrollo de aplicaciones de escritorio usando el lenguaje Java y un entorno de desarrollo integrado (IDE) para desarrollar bajo esta plataforma, pero también admite otros lenguajes de programación como C/C++, PHP, entre otros. Existe además un número importante de módulos para ser extendido. NetBeans IDE es un producto libre y gratuito sin restricciones de uso.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos.

La versión 6.9 incluye significativas mejoras y nuevas características, incluyendo una reestructuración completa de la infraestructura del editor, soporte para lenguajes adicionales, nuevas características productivas, y un proceso simplificado de instalación que permite instalar y configurar fácilmente el IDE según las necesidades del usuario.

El IDE ha sido desarrollado para distintas plataformas como Linux, MacOS X, Solaris y también Windows. (20)

1.15 Conclusiones parciales.

Luego de haber realizado un profundo estudio de las tecnologías existente para el desarrollo de aplicaciones web se seleccionaron las siguientes herramientas para darle solución a la situación existente la cual se utilizará PHP como lenguaje de programación, teniendo en cuenta algunas de sus ventajas como ser un lenguaje multiplataforma, con capacidad de conectarse con la mayoría de los manejadores de base de datos que se utilizan en la actualidad; y lenguaje del lado del cliente JavaScript, HTML y CSS. Como sistema gestor de base de datos PostgreSQL por su rapidez, confiabilidad y facilidad de uso. La metodología de desarrollo: RUP con notación UML debido a sus características y organización del trabajo. Herramienta CASE de Modelado UML: Visual Paradigm v6.4 y Servidor Web: Apache. Se propone NetBeans como herramienta para el desarrollo del sistema, pues es un producto gratis y de código abierto que ofrece a los desarrolladores numerosas ventajas en la creación de nuevas aplicaciones multiplataforma según las peticiones del usuario.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

1.1 Introducción.

En el presente capítulo se brinda una descripción de las principales características del sistema propuesto. Se define el modelo de dominio que permitirá una mayor comprensión de cada uno de los procesos que se automatizarán. Se identifican los actores y casos de uso, detallándose sus relaciones mediante el diagrama de caso de uso del sistema y las descripciones de estos. Además se realiza el levantamiento de los requerimientos funcionales y no funcionales que deberá poseerla aplicación.

1.2 Descripción de la solución propuesta.

A partir de los conocimientos obtenidos durante el proceso de investigación, se propone realizar el análisis, diseño e implementación de la técnica de apoyo al aprendizaje y enseñanza de la asignatura de Física, en el contenido de la Ley de Gauss, dicho sistema permite el control del proceso de asignación, resolución y evaluación de ejercicios orientados por los profesores a cada uno de sus estudiantes. El sistema consiste en una aplicación Web, diseñada principalmente para apoyar el modelo centrado en el auto aprendizaje. El mismo está dirigido a estudiantes de segundo año de la carrera de Ciencias Informática en la Facultad 5, permitiendo la autenticación por el LDAP logrando así la confiabilidad de la información y la seguridad del sistema propuesto, permite además la gestión de cada uno de los profesores que impartirán la asignatura, así como la creación de ejercicios, valoración automática de las respuestas dadas por los estudiantes, generación de reportes en formato pdf con todas las evaluaciones obtenidas por cada paso en cada intento de realización y la consulta previa o posterior de la documentación referente al tema por parte de los mismos.

Esta aplicación dota a los estudiantes de una herramienta acerca de un tema que actualmente presenta grandes dificultades. Uno de los principales problemas es la resolución de este tipo de ejercicios siguiendo la secuencia de pasos definidos para su resolución. Es por ello que a partir de diferentes encuentros, entrevistas y siguiendo la metodología seguida por los profesores de la asignatura en cada uno de los grupos, se definen a continuación los distintos pasos que el estudiante podrá desarrollar en el sistema para así aumentar su preparación individual.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

1. Identificar las regiones de análisis a partir de la delimitación de la superficie gaussiana y delimitación de los vectores.
2. Plantear la ley de Gauss.
3. Determinar el valor numérico del ángulo que forma la intensidad del campo eléctrico con el diferencial de superficie.
4. Determinar el valor de la carga neta (q_n) calculando la integral según el tipo de distribución.
5. Despejar la intensidad del campo eléctrico y calcular la integral del diferencial de superficie.
6. Determinar la expresión para el cálculo de la intensidad de campo eléctrico en la región indicada.
7. Indicar la unidad de medida correspondiente.

Actualmente la aplicación cuenta con dos ejercicios registrados, los cuales fueron tomados de las clases prácticas referentes al tema, que se encuentran en el Entorno Virtual de Aprendizaje (EVA). Los enunciados se anexan a la investigación.

1.3 Modelo de dominio.

El modelo de dominio tiene como objetivo describir y comprender las clases y objetos más significativos dentro del contexto del problema, lo cual ayuda a definir los procesos y roles para el sistema a desarrollar, facilitándole a los usuarios, desarrolladores y clientes el manejo de un vocabulario común, que les permita el entendimiento del problema a automatizar.

Al sistema a informatizar, se le realiza modelo de dominio debido a que no se puede realizar Modelo de Negocio pues resulta complejo identificar claramente los procesos que pertenecen al mismo debido a que no están bien estructurados. Ello hace que se presente una vista conceptual de forma general de la aplicación para capturar los objetos más importantes, los eventos que suceden en el entorno donde estará el sistema y la relación existente entre ellos.

A continuación se muestra el glosario de términos que describe los conceptos fundamentales relacionados con el proceso a automatizar:

- **Estudiante:** representa a todos los estudiantes de la Facultad 5 que reciben la asignatura de Física y que a la vez son evaluados por los profesores.
- **Profesor:** es la persona encargada de dar a conocer a sus estudiantes la asignación de los ejercicios de cada cual, además de supervisar y documentar los resultados de la realización de este.

- **Clases Prácticas:** tipo de clase en la que el estudiante es el mayor protagonista, respondiendo en la libreta un grupo de ejercicios que han sido orientados.
- **Registro de Evaluaciones:** medio en el cual el profesor va conservando cada una de las evaluaciones realizadas a los estudiantes en cada uno de los contenidos impartidos.
- **Materiales de apoyo a la docencia:** conjunto de libros, conferencias, guías de estudio que contienen información referente a las temáticas impartidas en la asignatura de Física.

1.3.1 Diagrama de clases del dominio.

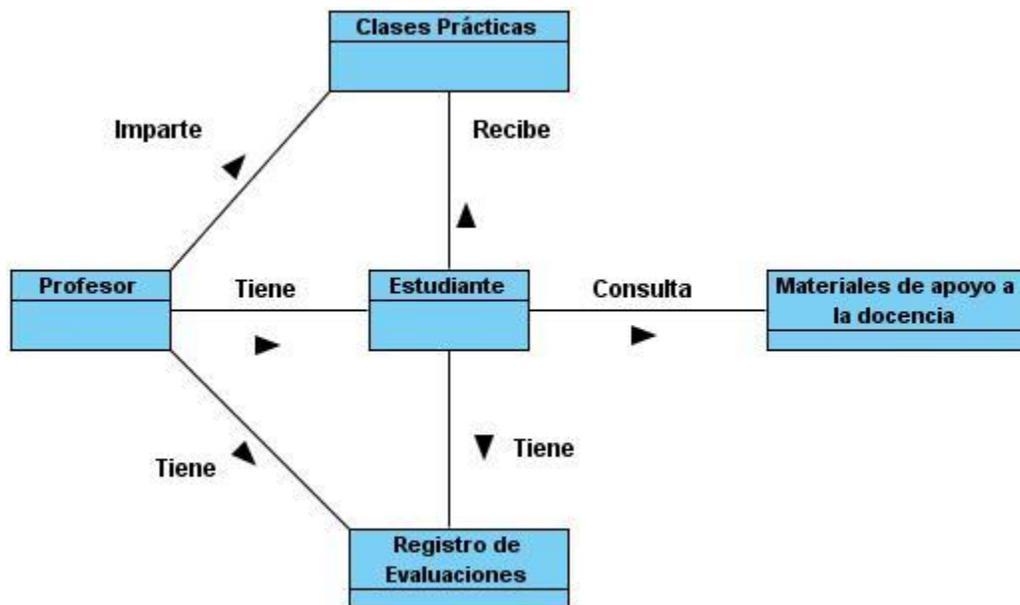


Ilustración 1: Representación del Modelo de Dominio.

1.4 Especificación de los requisitos didácticos y de software.

1.4.1 Requerimientos Funcionales.

Los requisitos funcionales son las acciones que los usuarios realizarán y el sistema debe ser capaz de responder a ellas. La IEEE-STD-729 (Institute of Electrical and Electronics Engineers) los define como la condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo.

Otras de las definiciones más consultada es la de Agudelo (2007) que plantea que un requisito funcional es la condición o capacidad que debe ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

A continuación se enuncian los requerimientos funcionales que debe cumplir el sistema:

| No. RF | Nombre del RF | Descripción |
|---|---|---|
| RF. 1 | Autenticar usuarios. | El sistema debe ser capaz de solicitar al cliente su nombre de usuario y contraseña para poder acceder al sistema, el que debe hacer una validación verificando si son correctos los datos entrados usando el servicio ldap. La aplicación producto al usuario autenticado debe permitir el acceso a las funcionalidades de acuerdo a los privilegios otorgados. |
| RF. 2 | Cerrar sesión. | El sistema debe permitir al usuario autenticado poder finalizar su sección de trabajo en cualquier momento y desde cualquier estación de trabajo. |
| RF. 3 RF. 3.1 RF. 3.2 RF. 3.3 | Gestionar profesores. <ul style="list-style-type: none">• Adicionar profesor.• Eliminar profesor.• Modificar profesor. | La aplicación debe permitir al profesor jefe de la asignatura de Física poder adicionar, modificar y/o eliminar un nuevo profesor al claustro de la asignatura. |
| RF. 4 RF. 4.1 RF. 4.2 RF. 4.3 | Gestionar grupo. <ul style="list-style-type: none">• Adicionar grupo.• Eliminar grupo.• Modificar grupo. | La aplicación debe permitir al profesor jefe de la asignatura de Física poder crear un nuevo grupo docente, modificar los valores con los que fueron registrados, además de eliminarlos en caso de desaparecer. |
| RF. 5 RF. 5.1 RF. 5.2 RF. 5.3 RF. 5.4 | Gestionar ejercicio. <ul style="list-style-type: none">• Adicionar ejercicio.• Modificar ejercicio.• Eliminar ejercicio.• Listar ejercicio. | La aplicación debe permitir al jefe de la asignatura crear nuevos ejercicios. Para ello debe llenar la siguiente estructura: Enunciado del ejercicio y la imagen correspondiente (en caso de tenerla), poder modificar los valores entrados en la creación de un ejercicio, ya sea para incorporarle o quitarle información, eliminarlos con sus respectivas respuestas y listar todos los ejercicios registrados en el sistema. |
| RF. 6 RF. 6.1 | Gestionar respuestas. <ul style="list-style-type: none">• Adicionar respuestas. | 6.1 La aplicación debe permitir al jefe de la asignatura poder insertar las posibles respuestas para cada una de |

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

| | | |
|----------------|--|--|
| RF. 6.2 | <ul style="list-style-type: none">• Eliminar respuestas. | las regiones que serán analizadas en los ejercicios, además de poder eliminar las respuestas dadas en cada una de las regiones que fueron analizadas. |
| RF. 7 | Asignar ejercicio. | El sistema debe permitir a los profesores poderle asignar a sus estudiantes los ejercicios a realizar a partir de los distintos niveles de complejidad. |
| RF. 8 | Realizar ejercicio | La aplicación debe permitir que el estudiante responda los ejercicios orientados por su profesor, determinando paso a paso la solución, además debe permitir al estudiante conocer la evaluación obtenida que se le mostrará al profesor mediante un reporte generado, con cada uno de los intentos completados. |
| RF. 9 | Generar reporte | La aplicación debe permitir al profesor y jefe de la asignatura de Física poder generar un reporte en formato .pdf por cada estudiante de su grupo, con cada una de las evaluaciones en los diferentes pasos durante el desarrollo de los ejercicios. |
| RF. 10 | Consultar bibliografía | La aplicación debe permitir a los estudiantes poder consultar alguna documentación referente al tema que se está evaluando en la aplicación para así poder consolidar lo aprendido. |

1.4.2 Requerimientos no Funcionales.

Los requerimientos no funcionales son las propiedades o cualidades que el producto debe tener, imponen restricciones en el diseño o la implementación. Deben pensarse en estas propiedades como las características que hacen al producto atractivo, usable, rápido o confiable. Por tanto son los criterios que se deben juzgar para determinar el funcionamiento de un sistema, verificando cómo este debería ser. (21)

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

A continuación se muestran los que debe cumplir el mismo.

| | | |
|---------------|---------------------------------------|---|
| RnF. 1 | Apariencia o interfaz externa. | El sistema diseñado debe tener una interfaz fácil de usar y amigable, para llamar la atención del usuario con una navegación sencilla principalmente. |
| RnF. 2 | Usabilidad. | El sistema podrá ser utilizado por personas que tengan un conocimiento básico en el manejo de las computadoras. El software tendrá siempre visible la opción de Ayuda, lo que posibilitará una mejor explotación de sus funcionalidades por parte de los usuarios. El sistema estará disponible las 24 h del día en un servidor destinado por la facultad para su implantación. |
| RnF. 3 | Rendimiento. | Los tiempos de respuesta y velocidad de procesamiento de la información serán rápidos, no mayores de 5 segundos para las actualizaciones y 20 para las recuperaciones. |
| RnF. 4 | Seguridad. | Los usuarios deberán autenticarse antes de entrar al sistema. El sistema deberá garantizar el acceso controlado a la información. Este debe influir sobre cómo se presentan las interfaces para cada usuario dependiendo del nivel de acceso a la información. |
| RnF. 5 | Software. | <u>Para el cliente:</u> Utilizar cualquier navegador (Mozilla Firefox, Internet Explorer, entre otros). Sistema operativo Windows XP o superior, o Linux. <u>Para el servidor:</u> Sistema operativo Windows o Linux en cualquiera de sus distribuciones. Un servidor Apache con módulo PHP disponible. Un servidor de bases de datos PostgreSQL. |
| RnF. 6 | Hardware. | Capacidad de Disco Duro igual o superior a 5 Giga bytes. Se requiere un mínimo de 256 MB de RAM y 1.6 GHz de velocidad de procesamiento. |

1.4.3 Requerimientos didácticos.

El Modelo Pedagógico de la carrera de Ingeniería en Ciencias Informáticas en la UCI se caracteriza por ser un modelo centrado en el aprendizaje presentando dos ciclos, el ciclo básico desde el primer semestre de la carrera hasta el quinto y un ciclo profesional desde el sexto semestre hasta el décimo.

La asignatura de Física se imparte en el segundo año de la carrera, por lo que los estudiantes al iniciar su estudio han dejado de consolidar sus habilidades en la materia por 1 o 2 años, por lo que se les dificulta la adquisición de nuevos conocimientos; es por ello que un sistema informático dirigido para el apoyo del aprendizaje debe estar diseñado de manera que propicie, entre otros aspectos, que el estudiante vea de manera permanente las relaciones de conjunto en el sistema de las ciencias, el auto diagnóstico, el desarrollo de actividades conjuntas y que conduzca al desarrollo del estudiante.

Atendiendo a ello se asumen las recomendaciones didácticas planteadas por la MSc. Yeily Delgado (2011) para utilizar las TIC dentro del PEA, estas son:

- Considerar las tareas docentes como un elemento fundamental que contribuya a la apropiación reflexiva de los contenidos.
- Estimular en los estudiantes la solución de problemas que posibiliten el desarrollo interdisciplinario de la asignatura.
- Desarrollar actividades propiciando el uso de las tecnologías que garanticen la atención diferenciada a los estudiantes y la interdisciplinariedad.

1.5 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. (21)

1.5.1 Actores del sistema.

Los actores del sistema suelen ser los trabajadores del negocio. Son las personas o sistemas que van a interactuar directamente con la aplicación con el objetivo de beneficiarse de sus funcionalidades.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

| Actor del sistema | Descripción |
|--|---|
| Usuario | Generalización de los actores: Estudiante y Profesor que son los encargados de asignar y responder cada uno de los ejercicios presentes en la aplicación. |
| Jefe de la Asignatura de Física | Es la persona encargada de registrar cada uno de los profesores que impartirán la asignatura, así como asignarle el o los grupos que atenderán, además de registrar los ejercicios en la aplicación y sus respectivas respuestas. |
| Profesor | Es la persona encargada de dar a conocer a sus estudiantes la asignación de ejercicios de cada cual, además de supervisar y documentar los resultados de la realización de estos. |
| Estudiante | El actor en cuestión representa a todos los estudiantes de la Facultad 5 que reciben la asignatura de Física y la vez son evaluados por los profesores. Son los encargados de resolver los ejercicios que les fueron orientados. |

1.5.2 Diagrama de casos de uso del sistema:

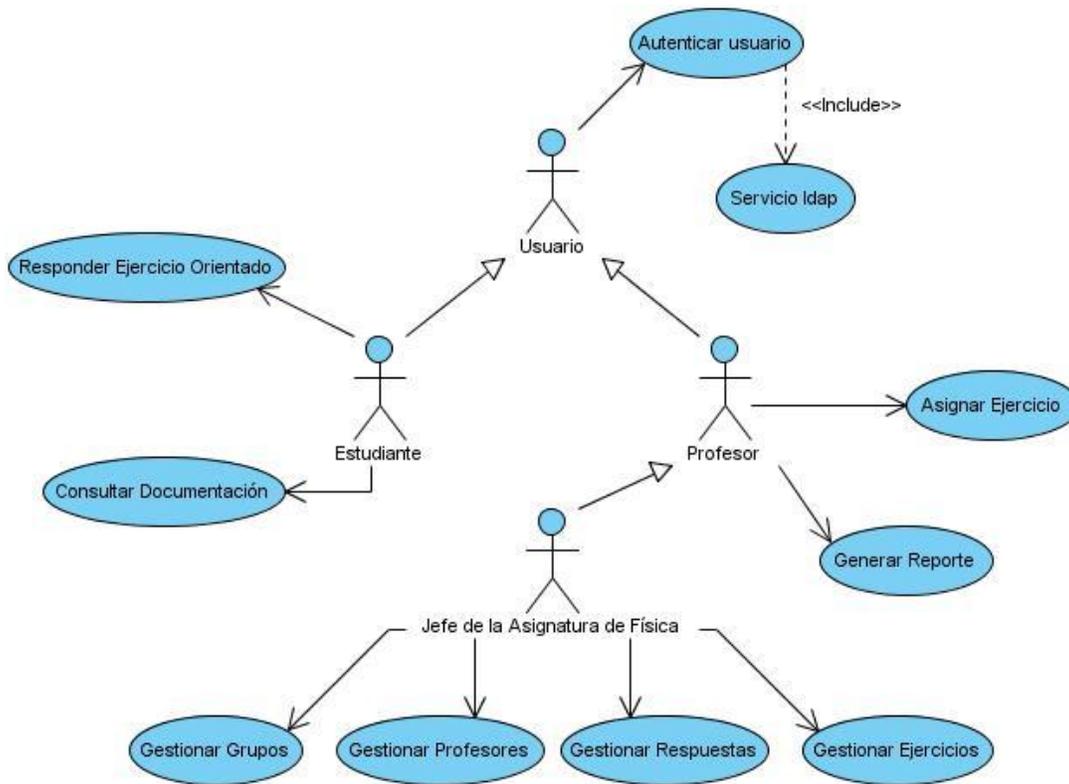


Ilustración 2: Representación del Modelo de Casos de Uso del Sistema.

1.5.3 Descripción de los casos de uso del sistema.

Un caso de uso es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso. En el contexto de ingeniería de software, un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. (22)

Se muestran a continuación los casos de uso más significativos los demás constituyen anexos.

Tabla 1 Especificación de caso de uso: Gestionar Ejercicio

| | |
|---------------------|--|
| Caso de Uso: | CU_ Gestionar ejercicio. |
| Objetivo | Gestionar cada uno de los ejercicios que serán realizados por los estudiantes. |
| Actores | Jefe de la asignatura de Física. |

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

| | | |
|---|--|--|
| Resumen | El caso de uso inicia cuando el Jefe de la asignatura de Física lleva a cabo todo el proceso de gestión de los distintos ejercicios ya sea adicionar, eliminar y/o modificar uno de ellos. | |
| Complejidad | Media. | |
| Requisitos de Referencia | RF 5.1, 5.2, 5.3 | |
| Prioridad | Crítico. | |
| Precondiciones | El actor debe estar autenticado como Jefe de la asignatura de Física previamente. | |
| Flujo de eventos | | |
| Flujo básico “Escenario adicionar” | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Selecciona la opción Gestionar Ejercicios del menú principal. | <p>1.1 Muestra el formulario Adicionar nuevo ejercicio con los siguientes campos:</p> <ul style="list-style-type: none"> • Enunciado del ejercicio. • Subir imagen del enunciado del ejercicio al servidor (<i>campo opcional</i>). <p>1.2 Muestra además el botón Ejercicios que muestra el listado de todos los ejercicios registrados en la aplicación.</p> |
| 2. | Llena el campo requerido para el Enunciado del ejercicio y presiona Enviar. (Alterno 1y Alterno 3). | <p>2.1 Verifica que el campo Enunciado del ejercicio no este vacío.</p> <p>2.2 Verifica en la BD que el ejercicio no existe.</p> <p>2.3 Retorna al formulario Adicionar nuevo ejercicio mostrando un mensaje de éxito de la operación: “Ejercicio</p> |

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

| | | registrado con éxito”. 2.4 Termina el caso de uso. |
|---------------------------------------|--|--|
| Flujos alternos (Escenario adicional) | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | En el paso 2 del flujo normal de eventos presiona clic en la opción de subir una imagen al servidor y presiona Enviar. | 1.1 Verifica que la ruta de la imagen existe. 1.2 Sube la imagen al servidor. (Alterno 2) 1.3 Registra en la BD la ruta donde fue guardada la imagen. |
| 2. | En el paso 2 del flujo normal de eventos presiona Enviar con una ruta para la imagen que no existe. | 2.1 Muestra un mensaje de error indicando: “No se ha podido subir el fichero”. |
| 3. | En el paso 2 del flujo normal de eventos presiona el botón Enviar teniendo aun el campo Enunciado del ejercicio vacío. | 3.1 Muestra un mensaje de error indicando: “No se permiten campos vacíos”. 3.2 El sistema retorna al formulario principal. |

Prototipo de interfaz

Enunciado del ejercicio.

Subir imagen del enunciado del ejercicio al servidor

Examinar...

Enviar >

Ejercicios >

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

| Flujo de eventos | | |
|--------------------------------------|---|--|
| Flujo básico “ Escenario modificar ” | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Presiona el botón Ejercicios en el escenario Adicionar nuevo ejercicio. | 1.1 Muestra una interfaz donde son listados todos los ejercicios con sus respectivas imágenes en caso de tenerlas que pueden ser modificados. |
| 2. | Presiona el botón  presente en el lateral derecho del ejercicio que desea modificar sus datos. | 2.1 Muestra la interfaz Modificar Ejercicio con los valores existentes en la base de datos en cada uno de los campos. |
| 3. | Modifica los datos pertinentes y presiona el botón Enviar. (Alternó 1, Alternó 2 y Alternó 3 del Escenario Adicionar nuevo ejercicio). | 3.1 Verifica que todos los campos estén llenos. 3.2 Verifica en la BD que el enunciado del ejercicio no existe (<i>en caso de ser modificado ese campo</i>). 3.3 Retorna al formulario Listado de ejercicios mostrando un mensaje de éxito de la operación: “ Grupo modificado con éxito ”. 3.4 Termina el caso de uso. |
| Flujo de eventos | | |
| Flujo básico “ Escenario eliminar ” | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Presiona el botón Ejercicios en el escenario Adicionar nuevo ejercicio. | 1.1 Muestra una interfaz donde son listados todos los ejercicios con sus respectivas imágenes en caso de tenerlas que pueden ser eliminados. |
| 2. | Presiona el botón  presente en el | 2.1 Elimina la asignación que tuvo algún estudiante del ejercicio, las respuestas |

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

| | | |
|-----------------------|--|--|
| | lateral derecho del ejercicio que se desea eliminar. | que dio del mismo. Además elimina todos los datos de la base de datos del ejercicio seleccionado. 2.2 Retorna a la interfaz que muestra el listado de ejercicios registrados. |
| Poscondiciones | Se le muestra al usuario cada una de las interfaces de adicionar, modificar o eliminar un ejercicio. | |

Tabla 8 Especificación de caso de uso: Resolver ejercicio

| | | |
|--|---|--|
| Caso de Uso: | CU_ Resolver ejercicio. | |
| Objetivo | Llevar a cabo el proceso de las respuestas registradas por los estudiantes en cada uno de los ejercicios que se le son asignados. | |
| Actores | Estudiante. | |
| Resumen | El caso de uso inicia cuando el estudiante comienza a resolver los ejercicios que le fueron orientados por su profesor. En el mismo podrá conocer todas las evaluaciones obtenidas durante la realización de varios intentos. | |
| Complejidad | Alta. | |
| Requisitos de Referencia | RF 8. | |
| Prioridad | Crítico. | |
| Precondiciones | El actor debe estar autenticado como Estudiante y acceder a la opción de Resolver Ejercicios. Tener al menos un ejercicio asignado. | |
| Flujo de eventos | | |
| Flujo básico "Resolver ejercicio" | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Selecciona la opción Resolver Ejercicios | 1.1 Muestra el formulario con el listado |

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

| | | |
|----|--|---|
| | del menú principal. | de los ejercicios que le fueron orientados y las opción <i>Resolver</i> . |
| 2. | Selecciona la opción Resolver dando clic en la imagen  presente en el lateral derecho del ejercicio a realizar. | 2.1 Muestra el formulario Resolver ejercicio con cada uno de los campos que le deberá proporcionar una respuesta, como son: <ul style="list-style-type: none">• Seleccionar el área a analizar.• Plantear la ley de Gauss.• Determinar el ángulo que forma la intensidad del campo eléctrico con el diferencial de superficie.• Valor de la carga neta• Despeje y cálculo de la integral del diferencial de superficie.• Determinar la expresión para el cálculo de la intensidad de campo eléctrico.• Indicar la unidad de medida. |
| 3. | Registra las respuestas indicadas para cada uno de los campos y presiona Enviar. (Alternó 1). | 3.1 Verifica que no exista ningún campo vacío. 3.2 Verifica en la base de datos que las respuestas indicadas en cada uno de los campos corresponden a las registradas por el profesor. 3.3 Retorna al formulario Resolver ejercicio mostrando en cada uno de los campos la siguiente imagen  para el caso de que el campo sea correcto y la imagen  para el caso incorrecto, mostrando además la evaluación final del intento a partir de las combinaciones |

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

| | | |
|--|--|---|
| | | de bien y mal presentes en la realización. 3.4 Termina el caso de uso. |
| Flujos Alternos (Escenario registrar) | | |
| No | Acción del actor | Respuesta del sistema |
| 1. | En el paso 3 del flujo normal de eventos el actor no selecciona una región a resolver. | 1.1 Retorna el formulario Registrar ejercicio mostrando un mensaje de error indicando: <i>“Debes seleccionar una región a resolver”</i> . |
| Poscondiciones | Se le muestra al usuario cada una de las interfaces de registrar una respuesta de ejercicio. | |

1.6 Conclusiones parciales

En este capítulo se describieron las características del sistema a desarrollar. Se expusieron los conceptos relacionados en el modelo de dominio para lograr comprender el funcionamiento del sistema. Se especificaron todos los requisitos funcionales y no funcionales del sistema donde se recogen las cualidades y capacidades que el sistema propuesto debe tener. Se identificaron los actores que intervienen. Se definieron además los casos de uso, que fueron descritos de forma detallada y reflejan las funcionalidades recogidas previamente en los requerimientos.

Capítulo 3 Análisis y Diseño del Sistema

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA.

3.1 Introducción

El presente capítulo ofrece una vista del análisis y diseño como parte de la solución al problema planteado, modelándose los artefactos necesarios que contribuyen a la implementación de la aplicación. Se define el modelo de análisis y el modelo de diseño, donde se muestran los diagramas de clases del análisis, así como los diagramas de clases del diseño, expresados en diagramas de clases con estereotipos web, el diagrama de despliegue y una breve descripción de la arquitectura y de los patrones arquitectónicos utilizados. Este capítulo también recoge el modelo físico y el modelo lógico de datos, que serán la plataforma para la construcción de la base de datos que soportará el sistema.

3.2 Descripción de la arquitectura.

La arquitectura de software es el conjunto de decisiones significativas sobre la organización de un sistema, la selección de los elementos estructurales y las interfaces que componen la aplicación, junto con su funcionalidad y navegabilidad. Describe los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente. Consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software para un sistema de información. El sistema que se propone presenta una arquitectura basada en el patrón arquitectónico Modelo Vista Controlador (MVC).(23)

3.2.1 Modelo Vista Controlador (MVC).

Este patrón se utiliza cuando es necesario modularizar la interfaz de usuario, las reglas de negocios y el control de eventos. El modelo administra el comportamiento y los datos del dominio de aplicación, responde a requerimientos de información sobre su estado (usualmente formulados desde la vista) y responde a instrucciones de cambiar el estado (habitualmente desde el controlador). Mantiene el conocimiento del sistema. No depende de ninguna vista y de ningún controlador. La vista maneja la visualización de la información mientras que el controlador interpreta las acciones del ratón y el teclado, informando al modelo y/o a la vista para que cambien según resulte apropiado. (23)

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

El uso del patrón Modelo Vista Controlador (MVC) es de gran importancia en el desarrollo de la aplicación ya que describe una forma de organizar el código en la misma separando los datos, la interfaz de usuario y la lógica en tres componentes distintos.

Este patrón debe su nombre a los tres elementos que lo conforman, el Modelo, la Vista y el Controlador. El Modelo administra la información de la aplicación, la Vista contiene todos los elementos que conforman la interfaz gráfica y el Controlador se encarga de separar y modificar el modelo y la vista, manejando la comunicación y la interacción entre ellos. En la siguiente figura se muestra a grandes rasgos el funcionamiento del patrón:

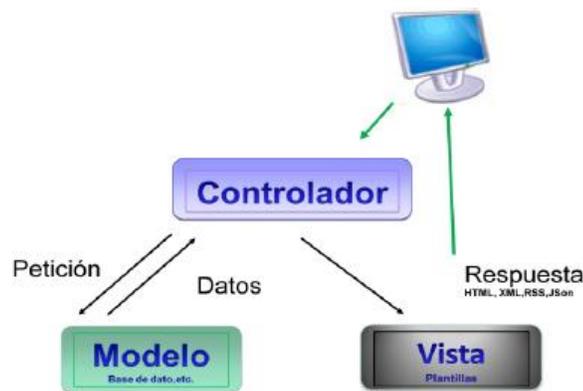


Ilustración 3: Representación del patrón MVC.

Cuando se recibe en el servidor una petición de un cliente a través de la red, esta es atendida por el Controlador, el que se encarga de acceder al Modelo para consultar, insertar o modificar datos y de devolver un resultado que se muestra en la Vista como respuesta a la petición.

Aunque el uso de este patrón aumenta ligeramente la complejidad del código, el mantenimiento de las aplicaciones se hace más sencillo, pues cuando se realiza un cambio en uno de sus componentes sólo se modifica el afectado. También es posible realizar cambios en un componente sin necesidad de conocer cómo funcionan los otros.

En el sistema en cuestión, la Vista va a estar constituida por el Paquete Presentación, el Controlador por el Paquete Control y el Paquete Clases Entidad y el Modelo por el Paquete Acceso a Datos.

3.3 Patrones de diseño

Los patrones de diseño que se utilizan son los patrones GRASP; por sus siglas en inglés (General Responsibility Assignment Software Patterns), los cuales tienen como objetivo la descripción de los principios fundamentales de diseño de objetos para la asignación de responsabilidades, y dentro de estos, los patrones Experto, Creador, Alta cohesión y Bajo acoplamiento. En el diseño de la

aplicación se utilizó principalmente el patrón Experto, el cual plantea que siempre se debe asignar una responsabilidad al experto en información, o sea, la clase con toda la información necesaria para llevarla a cabo, dichas clases son: CEstudiantes, CProfesor, CAsignatura.

El diseño obtenido cumple con los patrones de Bajo acoplamiento y Alta cohesión permitiendo la colaboración entre los elementos del diseño, sin verse afectados la reutilización de estos y el entendimiento cuando se encuentran aislados. La creación de clases controladoras facilitó realizar las operaciones del sistema, ya que no es factible manejarse en la capa de interfaz o presentación, debido a que estas operaciones reflejan los procesos de la empresa o dominio.

3.4 Modelo de análisis.

A través del modelado del análisis se obtiene una descripción detallada de los requisitos de software que ayudan a estructurar la vista interna del sistema y conformar la arquitectura del mismo. Este es descrito en el lenguaje de los desarrolladores pero sin tener en cuenta el lenguaje de programación ni la plataforma, ya que entre sus propósitos no está precisar cómo se implementará la solución, constituyendo así una aproximación al diseño a partir de los artefactos generados.

3.4.1 Diagramas de clases del análisis.

Los diagramas de clases del análisis permiten describir la estructura de un software a través de clases, atributos y las relaciones entre ellos, basándose en los conceptos significativos del dominio del problema.

Las clases del análisis se identifican por tres prototipos básicos:

- **Interfaz:** Modelan la interacción entre el sistema y sus actores.
- **Control:** Representan coordinación, secuencia, transacciones, y control de otros objetos. Representan derivaciones y cálculos complejos, como la lógica del negocio, que no pueden asociarse con ninguna información concreta, de larga duración, almacenada por el sistema.
- **Entidad:** Modelan la información que posee una larga vida y que es a menudo persistente. Además, modela el comportamiento asociado de algún fenómeno o concepto, como una persona o un objeto del mundo real.(24)

A continuación se muestran dos de los diagramas de clases de los análisis más significativos correspondientes a la solución que se propone. Los restantes diagramas constituyen anexos.

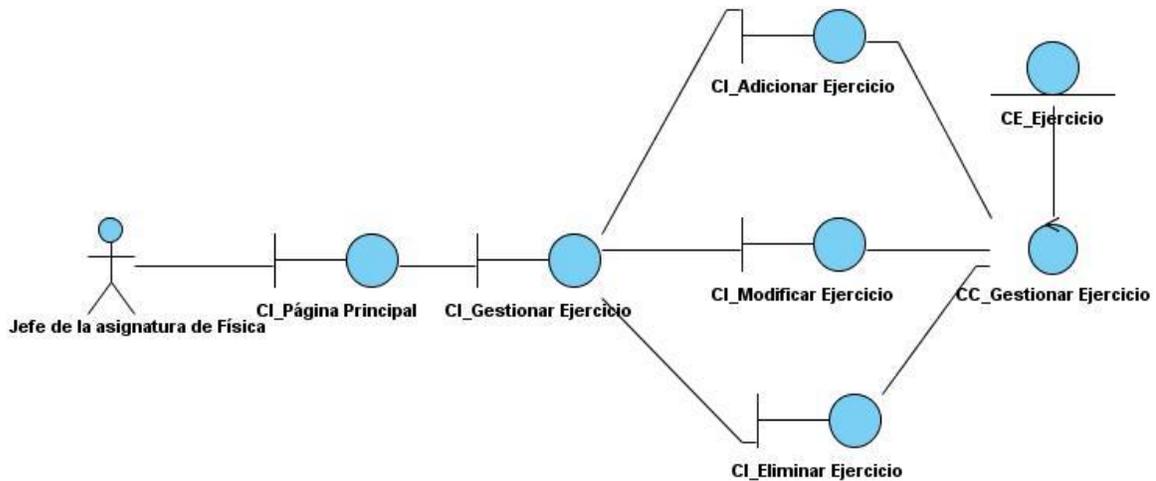


Ilustración 4: Diagrama de clases del análisis: CU_ Gestionar Ejercicio



Ilustración 5: Diagrama de clases del análisis: CU_ Resolver Ejercicio

3.4.2 Diagramas de Interacción (Colaboración).

Los diagramas de interacción se utilizan para modelar los aspectos dinámicos de un sistema, lo que conlleva a modelar instancias concretas o prototípicas de clases interfaces, componentes y nodos, junto con los mensajes enviados entre ellos, todo en el contexto de un escenario que ilustra un comportamiento.

UML define dos tipos de diagramas de interacción, los cuales sirven para expresar interacciones semejantes o idénticas del mensaje:

Diagrama de secuencia: es un diagrama que destaca el ordenamiento temporal de los mensajes. Muestran las interacciones expresadas en función de secuencias temporales.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Diagrama de colaboración: es un diagrama que destaca la organización estructural de los objetos que envían y reciben mensajes. Muestran las relaciones entre los objetos y los mensajes que intercambian.

A continuación se muestran dos de los diagramas de colaboración correspondientes a la solución que se propone. Los restantes diagramas constituyen anexos.

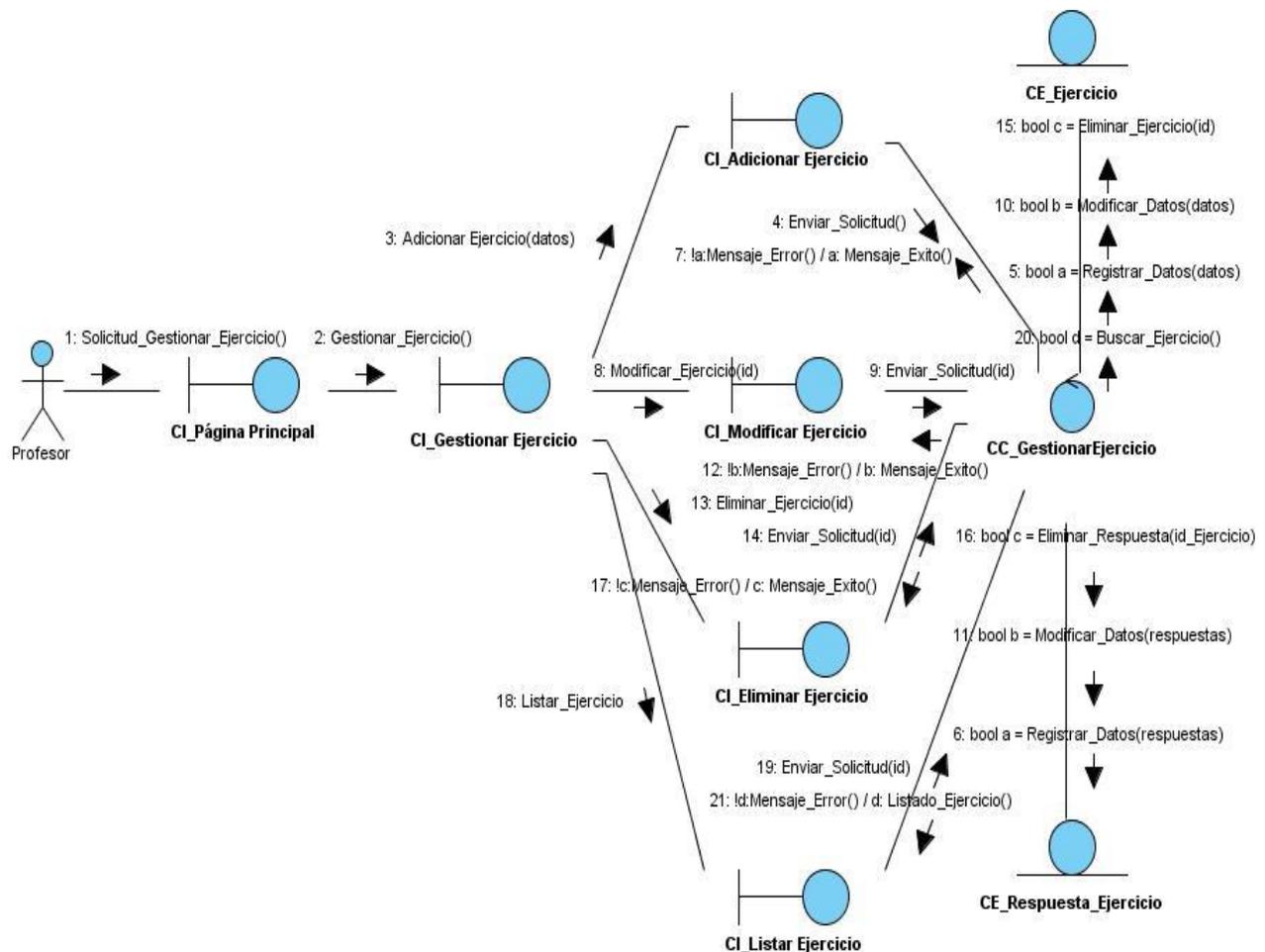


Ilustración 6: Diagrama de clases del análisis: CU_Gestionar Ejercicios

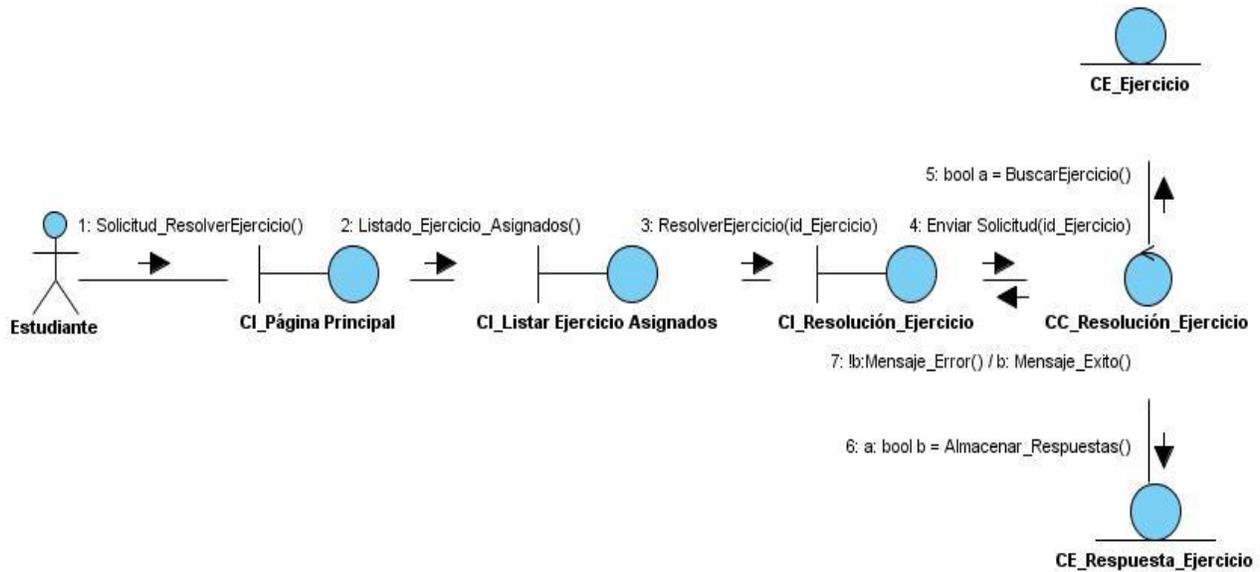


Ilustración 7: Diagrama de clases del análisis: CU_ Resolver Ejercicio

3.5 Modelo de diseño.

En el diseño del sistema se decide la estructura del mismo, a partir de la comprensión detallada de los requisitos de software que proporciona el modelo de análisis. Sirve de abstracción a la implementación del sistema y se utiliza como una entrada fundamental en las actividades de implementación. Durante esta etapa se modela la aplicación de forma tal que satisfaga todos los requisitos incluyendo los no funcionales.

3.5.1 Diagramas de clases del diseño con extensiones web.

Los diagramas de clases de diseño exponen un conjunto de interfaces, colaboraciones y sus relaciones. Se utilizan para modelar la vista de diseño estática de un sistema. Permiten visualizar, especificar y documentar modelos estructurales. Estos diagramas forman parte de las realizaciones de casos de uso.

Las clases del diseño están agrupadas en páginas servidoras que están compuestas por componentes Facelets, Rich Faces, JSF, Seam UI, así como código HTML. Todo este código será ejecutado en el servidor web, generando páginas clientes que pueden ser representadas por los navegadores web.

Las páginas clientes están compuestas por código HTML, CSS y JavaScript. Son interpretadas por los navegadores web, presentándole al usuario la interfaz con la que puede interactuar el sistema.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

Los formularios HTML son una sección de un documento que pueden contener elementos especiales llamados controles: casillas de verificación (*checkbox*), (*radiobutton*), menús, entre otros, además de rótulos (*labels*) en esos controles. Los usuarios normalmente completan un formulario modificando sus controles (introduciendo texto, seleccionando objetos de un menú, entre otros), y lo envían al servidor donde son procesados los datos enviados. Es una manera de obtener en el servidor información entrada por el usuario.

Las clases controladoras implementan la lógica del negocio que se está informatizando, generalmente cada una de estas se encarga de la implementación de un caso de uso o un proceso en dependencia de la complejidad de estos.

A continuación se muestran dos de las realizaciones de los diagrama de clases del diseño con estereotipos Web. Los restantes diagramas constituyen anexos.

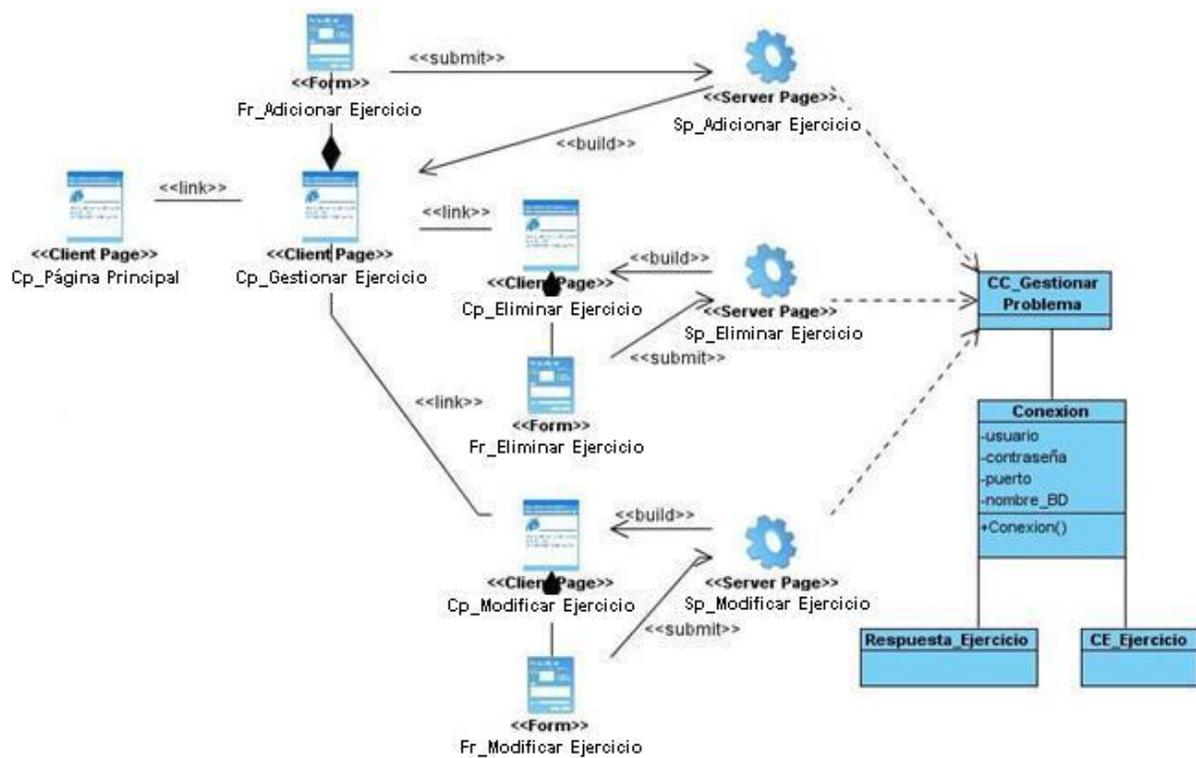


Ilustración 8: Diagrama de clases del diseño: CU_ Gestionar Ejercicio

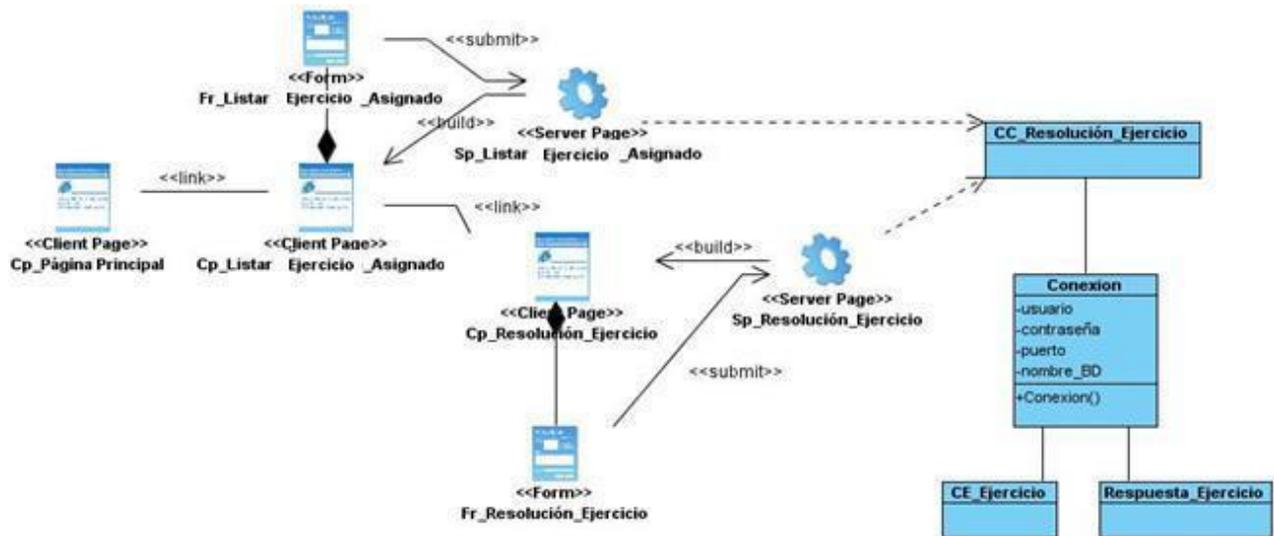


Ilustración 9: Diagrama de clases del diseño: CU_Resolver Ejercicio

3.6 Diseño de la Base de Datos.

Durante el diseño de la base de datos una de las técnicas más utilizadas para la creación de un sistema de información es el modelado entidad relación, mediante diagramas o modelos entidad relación (denominado por sus siglas, DER). Estos constituyen una representación conceptual de la información, mediante la cual se pretende visualizar los objetos que pertenecen a la base de datos como entidades (correspondiente al concepto de objeto de la Programación Orientada a Objetos) las cuales tienen diversos atributos relacionándose entre sí.

3.6.1 Modelo lógico de datos.

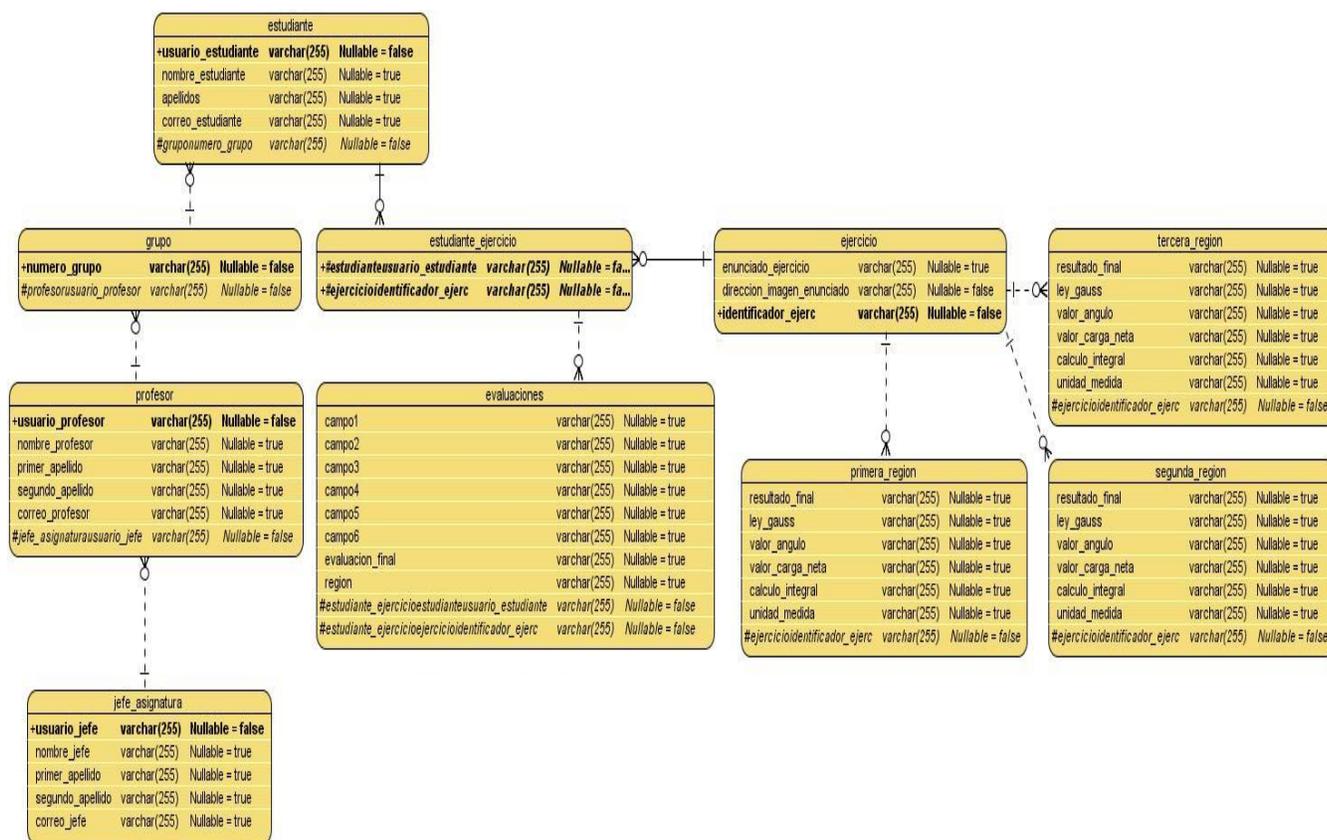


Ilustración 10: Modelo lógico de datos

3.6.2 Descripción de las tablas de la base de datos.

Se muestran a continuación las descripciones de las tablas de la base de datos, así como cada uno de sus atributos, tipo de datos y descripción de cada elemento.

| Nombre de la tabla: Profesor | | |
|--|---------|---|
| Descripción: Esta tabla es encargada de almacenar toda la información referente a los profesores que imparten la asignatura de Física en la facultad. | | |
| Atributo | Tipo | Descripción |
| usuario_profesor <i>(llave primaria)</i> | varchar | Representa el usuario asignado por la universidad a un profesor, el cual lo identificará. |
| nombre_profesor | varchar | Representa el nombre del profesor |
| primer_apellido | varchar | Representa el primer apellido del profesor |

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

| | | |
|--|---------|--|
| segundo_apellido | varchar | Representa el segundo apellido del profesor |
| correo_profesor | varchar | Representa el correo UCI asignado a un profesor en la universidad. |
| jefe_asignaturausuario_jefe (<i>llave foránea</i>) | varchar | Representa el identificador del jefe de la asignatura que lo dirige. |

Tabla 1: Descripción de la tabla Profesor presente en la base de datos.

| Nombre de la tabla: Estudiante | | |
|--|---------|---|
| Descripción: Esta tabla es encargada de almacenar toda la información referente a los estudiantes que reciben la asignatura de Física en la facultad. | | |
| Atributo | Tipo | Descripción |
| usuario_estudiante (<i>llave primaria</i>) | varchar | Representa el usuario asignado por la universidad a un estudiante, el cual lo identificará. |
| nombre_estudiante | varchar | Representa el nombre del estudiante. |
| apellidos | varchar | Representa los apellidos del estudiante. |
| correo_estudiante | varchar | Representa el correo UCI asignado a un estudiante en la universidad. |
| gruponumero_grupo (<i>llave foránea</i>) | integer | Representa el identificador del grupo al que pertenece ese estudiante. |

Tabla 2: Descripción de la tabla Estudiante presente en la base de datos.

| Nombre de la tabla: Jefe de Asignatura | | |
|---|---------|---|
| Descripción: Esta tabla es encargada de almacenar toda la información referente al jefe de la asignatura de Física en la facultad. | | |
| Atributo | Tipo | Descripción |
| usuario_jefe (<i>llave primaria</i>) | varchar | Representa el usuario asignado por la universidad al profesor jefe de la asignatura, el cual lo identificará. |
| nombre_jefe | varchar | Representa el nombre del profesor jefe de la asignatura. |

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

| | | |
|------------------|---------|--|
| primer_apellido | varchar | Representa el primer apellido del profesor jefe de la asignatura. |
| segundo_apellido | varchar | Representa el segundo apellido del profesor jefe de la asignatura. |
| correo_jefe | varchar | Representa el correo UCI asignado a un profesor en la universidad. |

Tabla 3: Descripción de la tabla Jefe de Asignatura presente en la base de datos.

| Nombre de la tabla: Grupo | | |
|---|---------|---|
| Descripción: Esta tabla es encargada de almacenar toda la información referente a todos los grupos en la facultad que reciben la asignatura. | | |
| Atributo | Tipo | Descripción |
| numero_grupo (<i>llave primaria</i>) | integer | Representa el número del grupo docente que recibe la asignatura, el cual lo identificará. |
| profesorusuario_profesor (<i>llave foránea</i>) | varchar | Representa el identificador del profesor que le imparte la asignatura de Física. |

Tabla 4: Descripción de la tabla Grupo presente en la base de datos.

| Nombre de la tabla: Ejercicio | | |
|---|---------|--|
| Descripción: Esta tabla es encargada de almacenar toda la información referente a todos los ejercicios subidos por los profesores. | | |
| Atributo | Tipo | Descripción |
| id_ejercicio (<i>llave primaria</i>) | varchar | Representa el identificador del ejercicio creado. |
| enunciado_ejercicio | varchar | Representa toda la descripción o enunciado de un ejercicio específico. |
| direccion_imagen_enunciado | varchar | Representa la imagen que acompañará el ejercicio |

Tabla 5: Descripción de la tabla Ejercicio presente en la base de datos.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

| Nombre de la tabla: Estudiante_ejercicio | | |
|--|---------|---|
| Descripción: Esta tabla es encargada de almacenar toda la información referente a las asignaciones de ejercicios que ha ido teniendo el estudiante. | | |
| Atributo | Tipo | Descripción |
| estudianteusuario_estudiante | varchar | Representa el usuario al cual se le está asignando un ejercicio. |
| identificador_ejerc (llave foránea) | integer | Representa el identificador del ejercicio que se le fue asignado. |

Tabla 6: Descripción de la tabla Respuesta Ejercicio presente en la base de datos.

| Nombre de la tabla: Evaluaciones | | |
|--|---------|--|
| Descripción: Esta tabla es encargada de almacenar toda la información referente a las evaluaciones de los intentos de resolución de los ejercicios encada uno de los pasos. | | |
| Atributo | Tipo | Descripción |
| campo1 | varchar | Representa el campo de la evaluación obtenida al plantear la ley de Gauss. |
| campo2 | varchar | Representa el campo de la evaluación obtenida al indicar el valor del ángulo formado en la representación gaussiana. |
| campo3 | varchar | Representa el campo de la evaluación obtenida al plantear la carga neta. |
| campo4 | varchar | Representa el campo de la evaluación obtenida al plantear el despeje de la intensidad del campo eléctrico y el cálculo de la integral formada. |
| campo5 | varchar | Representa el campo de la evaluación obtenida al plantear la expresión del cálculo de la intensidad de campo eléctrico. |
| campo6 | varchar | Representa el campo de la evaluación obtenida al indicar la unidad de medida. |
| evaluación_final | varchar | Representa el campo de la evaluación final de un intento para un ejercicio en una región |

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

| | | |
|---------------------------------------|---------|---|
| | | indicada. |
| id_ejer (llave foránea) | varchar | Representa el identificador del ejercicio que está siendo resuelto. |
| usuario_estudiante (llave foránea) | varchar | Representa el estudiante que realizó el ejercicio. |
| región | varchar | Representa la región que está siendo analizada y evaluada. |

Nombre de la tabla: Primera región

Descripción: Esta tabla es encargada de almacenar toda la información referente a la primera región que será analizada.

| Atributo | Tipo | Descripción |
|------------------|---------|--|
| id_ejercicio | varchar | Representa el identificador del ejercicio que se está resolviendo. |
| ley_gauss | varchar | Representa el valor ingresado para la ley de Gauss. |
| valor ángulo | varchar | Representa el valor ingresado para el ángulo formado en la representación gaussiana. |
| valor_carga_neta | varchar | Representa el valor ingresado para la carga neta indicada en la región. |
| calculo_integral | varchar | Representa el valor ingresado para el cálculo de la integral y despeje de la intensidad del campo eléctrico. |
| resultado_final | varchar | Representa el valor ingresado para el cálculo de la intensidad del campo eléctrico. |
| unidad_medida | varchar | Representa el valor ingresado para la unidad de medida correspondiente. |

Nombre de la tabla: Segunda región

Descripción: Esta tabla es encargada de almacenar toda la información referente a la segunda región que será analizada.

| Atributo | Tipo | Descripción |
|----------|------|-------------|
|----------|------|-------------|

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

| | | |
|------------------|---------|--|
| id_ejercicio | varchar | Representa el identificador del ejercicio que se está resolviendo. |
| ley_gauss | varchar | Representa el valor ingresado para la ley de Gauss. |
| valor ángulo | varchar | Representa el valor ingresado para el ángulo formado en la representación gaussiana. |
| valor_carga_neta | varchar | Representa el valor ingresado para la carga neta indicada en la región. |
| calculo_integral | varchar | Representa el valor ingresado para el cálculo de la integral y despeje de la intensidad del campo eléctrico. |
| resultado_final | varchar | Representa el valor ingresado para el cálculo de la intensidad del campo eléctrico. |
| unidad_medida | varchar | Representa el valor ingresado para la unidad de medida correspondiente. |

| Nombre de la tabla: Tercera región | | |
|--|-------------|--|
| Descripción: Esta tabla es encargada de almacenar toda la información referente a la tercera región que será analizada. | | |
| Atributo | Tipo | Descripción |
| id_ejercicio | varchar | Representa el identificador del ejercicio que se está resolviendo. |
| ley_gauss | varchar | Representa el valor ingresado para la ley de Gauss. |
| valor ángulo | varchar | Representa el valor ingresado para el ángulo formado en la representación gaussiana. |
| valor_carga_neta | varchar | Representa el valor ingresado para la carga neta indicada en la región. |
| calculo_integral | varchar | Representa el valor ingresado para el cálculo de la integral y despeje de la intensidad del campo eléctrico. |

| | | |
|-----------------|---------|---|
| resultado_final | varchar | Representa el valor ingresado para el cálculo de la intensidad del campo eléctrico. |
| unidad_medida | varchar | Representa el valor ingresado para la unidad de medida correspondiente. |

3.7 Diagrama de despliegue.

El modelo de despliegue representa un modelo de objetos que muestra cómo se distribuye físicamente el sistema, teniendo en cuenta la funcionalidad entre cada nodo de cómputo. Este modelo es utilizado como base para la realización de las actividades de diseño e implementación. Cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo hardware similar. Los nodos poseen relaciones que representan medios de comunicación entre ellos, tales como Internet, Intranet, bus y similares.

El modelo de despliegue puede describir diferentes configuraciones de red, incluidas las configuraciones para pruebas y para simulación, además de representar una correspondencia entre la arquitectura de software y la arquitectura del sistema (hardware).

La figura muestra el diagrama de despliegue del sistema que se propone.

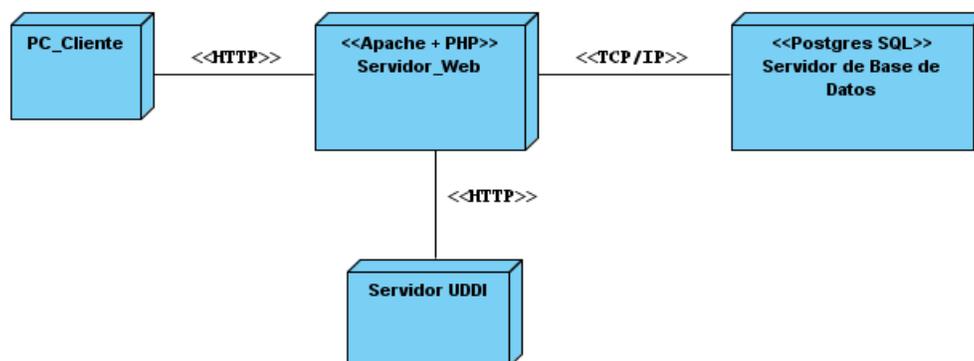


Ilustración 11: Modelo de Despliegue

El diagrama establece la relación existente entre los nodos:

Servidor UDDI: Servicios web que la UCI brinda para obtener información del personal de la universidad ya sea estudiantes, profesores o trabajadores que son autenticados en el dominio uci.cu mediante LDAP. El protocolo que utiliza es SOAP (Simple Object Access Protocol) por sus siglas en inglés.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

PC_Cliente: espacio de trabajo por el cual el cliente interactúa con la aplicación. Esta computadora debe tener instalado el Sistema Operativo Windows o GNU/ Linux, como navegador Web Internet Explorer o Mozilla Firefox 2.0 como mínimo.

Servidor Web: se utilizará el servidor Web apache más el lenguaje de programación PHP. Contiene toda la información referente a la aplicación.

Servidor Base de datos: se utilizará PostgreSQL como base de datos que permitirá almacenar toda la información de la aplicación.

3.8 Conclusiones parciales.

Luego de haber realizado la fase de análisis y diseño correspondiente para la elaboración del software que se propone, se transformaron los requerimientos en clases que definen cómo va a ser implementado el sistema, logrando adaptar el diseño para que coincida con el ambiente de implementación, diseñando el software con un enfoque hacia el rendimiento, se modelaron los artefactos del análisis y el diseño. Utilizando Visual Paradigm y UML, ambos brindaron una mejor visión sobre cómo llevar a cabo la implementación del sistema. La utilización de los patrones de asignación de responsabilidades (GRASP), para que el diseño tenga un Bajo acoplamiento, Alta cohesión, logrando con esto un mejor funcionamiento del sistema en cuanto a respuesta y eficiencia.

Capítulo 4 Implementación y Pruebas

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBAS.

1.1 Introducción.

En el presente capítulo se establecen criterios asociados a la implementación y validación del sistema propuesto. En la realización de diferentes sistemas informáticos juega un papel fundamental el uso de las técnicas de evaluación dinámica o pruebas. Para lograr esto debes llevar a cabo estrategias a la hora de evaluar dinámicamente el software partiendo del estudio y puesta en práctica del método de caja negra. Con este propósito se definieron los casos de pruebas afines con los procesos de negocio más significativos para la arquitectura. Además, se tienen los diagramas de componentes, lo que muestran las organizaciones y dependencias lógicas entre los elementos del software.

1.2 Diagrama de componentes.

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre los componentes del software, sean éstos componentes de código fuente, binarios o ejecutables. Desde el punto de vista del diagrama de componentes se tienen en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión del software, la reutilización, y las restricciones impuestas por los lenguajes de programación y las herramientas utilizadas en el desarrollo. Los elementos de modelado dentro de un diagrama de componentes serán componentes y paquetes.
(25)

A continuación se muestran los diagramas de componentes del sistema referentes a los requisitos funcionales: Gestionar ejercicio y Resolver ejercicio, los demás diagramas lo constituyen los anexos:

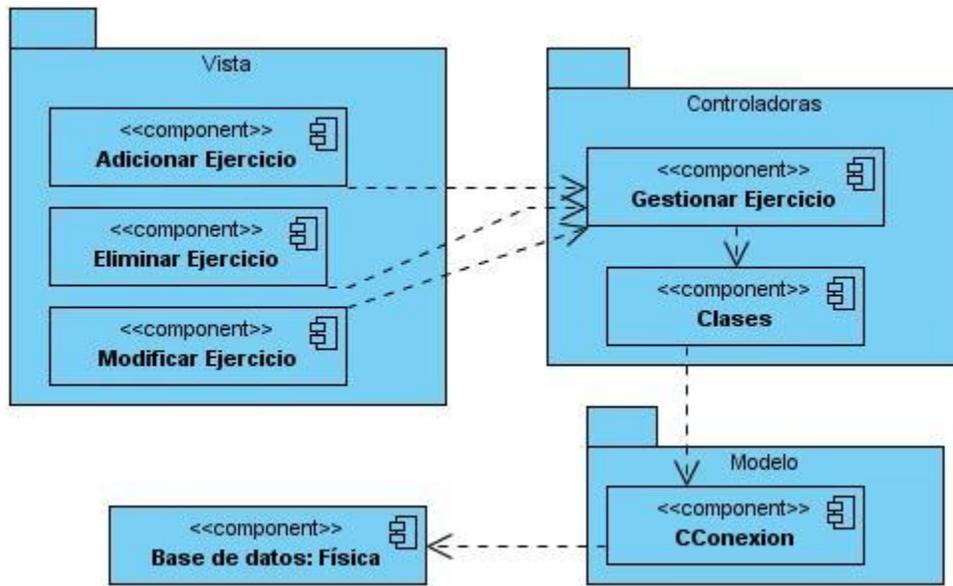


Ilustración 12: Diagrama de componentes: CU_ Gestionar ejercicio

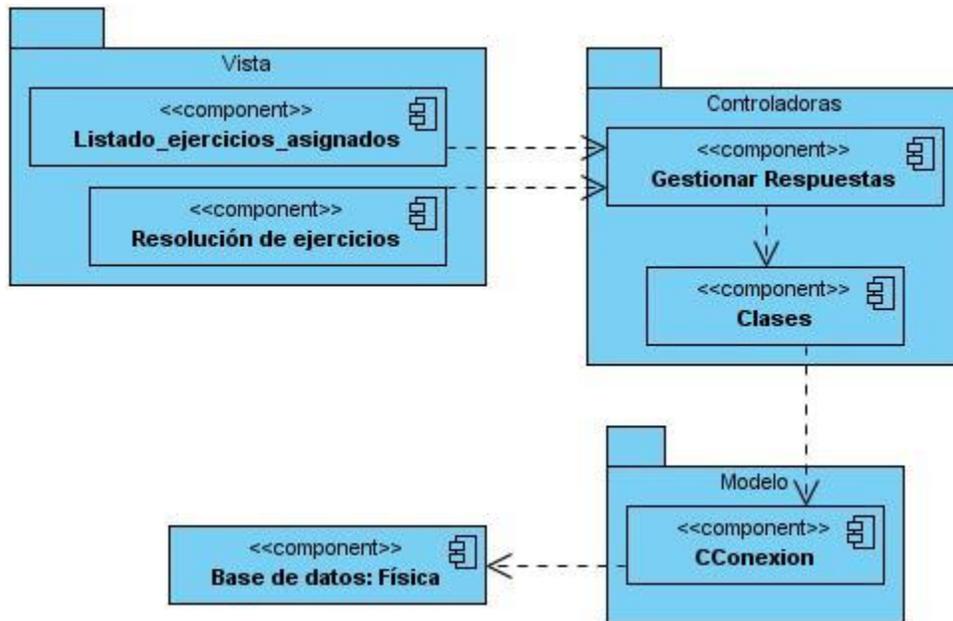


Ilustración 13: Diagrama de componentes: CU_ Resolver Ejercicio

1.3 Pruebas de software.

Las pruebas de software son un conjunto de herramientas, técnicas y métodos que evalúan el desempeño de un programa. Involucran las operaciones del sistema bajo condiciones controladas y evaluando los resultados, es por eso que la realización de las mismas a los software es un factor de vital importancia. Antes de liberar el producto es necesario tener la certeza de que este se encuentra libre de errores, aunque se considera que no se puede estar 100% seguro de esto. Pero una cosa sí es cierta, mientras más pruebas se le realicen al software más cerca se podrá estar de lograr un producto con la calidad esperada por los usuarios. De las pruebas se plantean varias normas que pueden servir acertadamente como objetivos de las mismas. Probar es un proceso de ejecución de un programa con la intención de descubrir un error. Una prueba tiene éxito si se descubre un error no detectado hasta entonces. (26)

Tipos de prueba

1. Pruebas de caja negra: Pruebas que se llevan a cabo sobre la interfaz del software. El objetivo es demostrar que las funciones del software son operativas, que las entradas se aceptan de forma adecuada y se produce un resultado correcto, y que la integridad de la información externa se mantiene (no se analiza el código).
2. Pruebas de caja blanca: Pruebas que se producen sobre el código del software. Es considerada como uno de los tipos de pruebas más importantes que se le aplican a los programas informáticos, logrando como resultado que disminuya en un gran por ciento el número de errores existentes en los sistemas y por ende, se logre una mayor calidad y confiabilidad.

1.3.1 Diseño de Casos de Prueba.

Caso de Prueba: Gestionar ejercicio.

| Escenario de la sección | Enunciado del ejercicio | Subir imagen | Respuesta del sistema | Flujo central |
|--|-------------------------|--------------|---|---|
| EC1.1 El campo enunciado del ejercicio en blanco. | I | N/A | Muestra un mensaje de error indicando: "Indique el enunciado del ejercicio". | 1. Al introducir los datos para el nuevo ejercicio se deja el campo enunciado del ejercicio vacío. 2. Se presiona el botón Enviar. |

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBAS.

| | | | | |
|--|-----|---|--|---|
| EC1.2Ejercicio creado con éxito. | V | V | Muestra un mensaje de éxito de la operación indicando: "Ejercicio registrado con éxito". | <ol style="list-style-type: none">1. Se introducen los datos correctamente.2. Se presiona el botón Enviar.3. Se muestra la notificación "Ejercicio registrado con éxito". |
| EC1.3Error al subir una imagen | N/A | I | Muestra un mensaje de error indicando: "Debe seleccionar la imagen". | <ol style="list-style-type: none">1. Se introducen los datos correctamente para el enunciado del ejercicio.2. Se selecciona la opción para subir imagen.3. Se presiona el botón Enviar sin indicar la ruta de la imagen. |
| EC1.4 Ruta del servidor para guardar la imagen. | N/A | I | Muestra un mensaje de error indicando: "La ruta de almacenamiento de la imagen no existe". | <ol style="list-style-type: none">1. Se introducen los datos correctamente para el enunciado del ejercicio.2. Se selecciona la opción para subir imagen.3. Se presiona el botón Enviar indicando la ruta de la imagen sin tener correctamente indicado donde será almacenada. |

Caso de Prueba: Resolver Ejercicio.

- 1: Selección de la región a analizar.
- 2: Plantear la ley de Gauss.
- 3: Determinar el valor numérico del ángulo que forma la intensidad del campo eléctrico con el diferencial de superficie.
- 4: despejar la intensidad del campo eléctrico y calcular la integral del diferencial de superficie.
- 5: Determinar la expresión para el cálculo de la intensidad de campo eléctrico en la región indicada.
- 6: Indicar la unidad de medida.

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBAS.

| Escenario de la sección | 1 | 2 | 3 | 4 | 5 | 6 | Evaluación Final. | Respuesta del sistema | Flujo central |
|---|-----|-----|-----|-----|-----|-----|-------------------|--|--|
| EC2.1 Deja al menos un campo en blanco | I | I | I | I | I | I | N/A | Muestra un mensaje de error indicando: "Debe llenar todos los campos". | <ol style="list-style-type: none"> 1. Al introducir los datos para la resolución del ejercicio se deja algún campo vacío. 2. Se presiona el botón Enviar. |
| EC2.2 Resolución correcta del ejercicio | V | V | V | V | V | V | V | Muestra un mensaje de éxito de la operación indicando que el ejercicio ha sido resuelto y muestra además la evaluación en el respectivo campo: "Ejercicio resuelto con éxito". | <ol style="list-style-type: none"> 1. Se introducen los datos en todos los campos. 2. Se presiona el botón Enviar. 3. Se muestra la notificación y la evaluación del ejercicio. "Ejercicio resuelto con éxito". |
| EC2.3 No selecciona la región a analizar | N/A | Muestra un mensaje de error indicando: "Debe seleccionar la región analizar". | <ol style="list-style-type: none"> 1. No se selecciona la región a analizar. 2. Se llenan todos los campos y se presiona el botón Enviar. 3. Se muestra la notificación de error indicando: "Debe seleccionar |

CAPÍTULO 4: IMPLEMENTACIÓN Y PRUEBAS.

| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|----------------------|
| | | | | | | | | | la región analizar”. |
|--|--|--|--|--|--|--|--|--|----------------------|

Los restantes casos de pruebas lo constituyen los anexos.

1.4 Resultados Obtenidos.

Para evaluar la solución se realizaron varias iteraciones con las que se probó el software íntegramente. RUP propone dos métodos fundamentales: caja blanca y caja negra. Se especifican las principales pruebas realizadas a la aplicación haciendo uso del método de caja negra, estas pertenecen al nivel de prueba de sistema y se detallan a continuación.

Pruebas Internas: Son realizadas con el fin de entregar un producto con la menor cantidad de errores posibles. Se centraron en el cumplimiento de las funcionalidades descritas en el listado de requerimientos y de casos de uso del sistema.

Pruebas Cruzadas: Fueron realizadas con el fin de encontrar la mayor cantidad de errores posibles en término de validaciones, pautas definidas por arquitectura de información, formato de los campos, entre otras.

A continuación se presentan los resultados arrojados durante las diferentes pruebas aplicadas:

Pruebas Internas:

| Módulos | Casos de Uso | No Conformidades | | | |
|----------------------|--------------|------------------|-------|------|-------|
| | | Alta | Media | Baja | Total |
| Autenticar Usuario | 2 | 0 | 1 | 2 | 3 |
| Gestionar Profesores | 3 | 1 | 1 | 2 | 4 |
| Gestionar Grupos | 4 | 0 | 0 | 1 | 1 |
| Gestionar Ejercicios | 5 | 0 | 1 | 0 | 1 |
| Gestionar Respuestas | 6 | 0 | 1 | 2 | 3 |
| Asignar Ejercicio | 7 | 0 | 0 | 1 | 1 |
| Evaluaciones | 8 | 0 | 0 | 1 | 1 |
| Resolver Ejercicio | 9 | 1 | 1 | 3 | 5 |

Pruebas Cruzadas:

| Módulos | Casos de Uso | No Conformidades | | | |
|----------------------|--------------|------------------|-------|------|-------|
| | | Alta | Media | Baja | Total |
| Autenticar Usuario | 2 | 0 | 0 | 2 | 2 |
| Gestionar Profesores | 3 | 1 | 0 | 2 | 3 |
| Gestionar Grupos | 4 | 0 | 0 | 2 | 2 |
| Gestionar Ejercicios | 5 | 0 | 1 | 1 | 2 |
| Gestionar Respuestas | 6 | 0 | 0 | 3 | 3 |
| Asignar Ejercicio | 7 | 0 | 0 | 1 | 1 |
| Evaluaciones | 8 | 0 | 0 | 1 | 1 |
| Resolver Ejercicio | 9 | 1 | 2 | 1 | 4 |

Una vez detectadas estas no conformidades incluyendo los casos de pruebas presentes en los anexos, se llegó al consenso de erradicarlas en un intervalo de 5 días. Para la siguiente iteración se corrigieron todos los errores existentes y no se encontraron nuevas fallas.

1.5 Conclusiones parciales.

En el desarrollo de este capítulo se lograron definir los artefactos correspondientes a la fase de la implementación como son los diagramas de componentes de cada caso de uso analizado. Se ejecutaron las pruebas de software, lo que permitió el posterior análisis de los resultados obtenidos en dos de los requisitos funcionales que el sistema debe cumplir. Además se solucionaron las no conformidades detectadas, posibilitando que el sistema funcione con la calidad esperada.

Conclusiones Generales

CONCLUSIONES GENERALES.

Al concluir el presente trabajo se arriba a las siguientes conclusiones:

- La investigación realizada sobre el estado actual del nivel de conocimiento que presentan los estudiantes de segundo año referente al tema de la Ley de Gauss permitió obtener la información que sirvió como base para confeccionar la solución propuesta.
- Se realizó el análisis, diseño e implementación de la propuesta del sistema informático, siguiendo el ciclo de vida que propone RUP como metodología de desarrollo, donde se definieron los conceptos necesarios para la formalización e implementación de la misma.
- Se logró la implementación de la solución propuesta cumpliendo con el objetivo general planteado.



Recomendaciones

RECOMENDACIONES

- Se recomienda la implementación de un módulo que se encargue de gestionar los usuarios que se clasifican como repitentes.
- Agregarle varios módulos que abarquen todos los contenidos tratados en el programa analítico de la asignatura de Física.
- Ampliar el alcance de la solución propuesta, haciéndola extensiva a todas las facultades de la universidad.
- Integrar la solución propuesta como uno de los materiales de apoyo para la asignatura de Física en el EVA.

Referencias Bibliográficas.

1. **A. Fajer, V y B. Henríquez.** *l despegue de la Física en Cuba desde 1959 hasta la década de los setenta: un enfoque abarcador.* s.l. : Berlín: Max Planck Institute for the History of Science, 2005. pág. Pág. 2.
2. *Un punto de vista acerca del proceso de Enseñanza-Aprendizaje en la Educación Física.* **Mola Cantero, Lic. Maribel y Sánchez Balbín, MSc. Lourdes Guillermina.** 2010.
3. **María A. Mendoza Sanchez.** Metodologías De Desarrollo De Software. [En línea] 7 de Junio de 2004. http://www.informatizate.net/articulos/metodologias_de_desarrollo_de_software_07062004.html.
4. **Sosa López, Dailyn .** Casos de Uso del Sistema. [En línea] 2009. <http://www.eumed.net/libros/2009c/585/Modelo%20de%20Casos%20de%20Uso%20del%20Sistema.htm>.
5. **RUMBAUGH, I.J.G.B.J.** *El Proceso Unificado de Desarrollo de Software.* s.l. : Addison Wesley, 2000.
6. Ecured: Requisitos no funcionales. [En línea] 2011. http://www.ecured.cu/index.php/Requisitos_no_funcionales.
7. **Más Rodés , Lic. Raúl .** La Informática en el proceso de enseñanza-aprendizaje de la Física. Alternativa metodológica para su utilización. [En línea] <http://www.ilustrados.com/tema/12222/Informatica-proceso-ensenanza-aprendizaje-Fisica-Alternativa.html>.
8. **I Delgado, M Hernández, M Sánchez, B Sánchez, O Chavianod y B Companionie.** UNA APLICACIÓN INFORMÁTICA PARA LA ENSEÑANZA DE LA FÍSICA. [En línea] 2011. <http://www.fisica.uh.cu/biblioteca/revcubfi/2011/Vol.28-No.1E/RCF-28-1E-2011-104.pdf>.
9. **Gras Martí, Albert , Cano Villalba, Marisa y Soler Selva, Vicent F.** Uso de las NTIC en la enseñanza de la física. [En línea] <http://agm.cat/eao/Ntic/msCINE.htm>.
10. **Romero, Hugo .** Rational Unified Process. [En línea] 2009. <http://www.slideshare.net/genesys/diseo-de-sistemas>.
11. **Rumbaugh, James.** El Lenguaje Unificado de Modelado (UML), Manual de Referencia. [En línea] <http://www.intercambiosvirtuales.org/libros-manuales/el-lenguaje-unificado-de-modelado-uml-de-james-rumbaugh-manual-de-referencia>.
12. Bizagi BPMN 2.0. [En línea] <http://www.bizagi.com/docs/BPMNbyExampleSPA.pdf>.
13. Ecured: Herramienta CASE. [En línea] http://www.ecured.cu/index.php/Herramienta_CASE.

14. **Hernández, Prof: Leydis Dayana.** Introducción a las Bases de Datos. [En línea]
<http://campus.dokeos.com/courses/BDFATLA/document/Introd.alasBasesdedatos.pdf?cidReq=BDFATLA>.
15. Investigación sobre PostgreSQL. [En línea] <http://es.scribd.com/doc/36570462/postgreSQL-investigacion>.
16. **FÉLIX, A. D. C. S.** El servidor de web Apache: Introducción práctica. Apache 1.x y 2.0 alpha. [En línea] 2000.
<http://acs.barrapunto.org/articulos/trunk/LinuxActual/Apache/apache.pdf>.
17. **Valdeli, Ilario.** Elementos de javascript. [En línea] 2006.
http://www.htmlpoint.com/javascript/corso/js_02.htm..
18. **Ajona, Guillermo Prado.** Página Web sobre HTML y CSS. [En línea]
<https://belenus.unirioja.es/~guprado/pagweb/caracss.html>.
19. Procesamiento del lado del servidor. [En línea] 2009.
<http://prograweb.com.mx/pweb/0301clienteServidor.html>.
20. **Sosa López, Dailyn .** [En línea]
<http://www.eumed.net/libros/2009c/585/Modelo%20de%20Casos%20de%20Uso%20del%20Sistema.htm>.
21. **Mamani Poma, Orlando Jimmy.** Desarrollo de un Sistema Informático para la Administración de Expedientes de la Universidad José Carlos Mariátegui. [En línea] <http://www.monografias.com/trabajos82/informe-practicas-preprofesionales-ing-sistemas-e-informatica/informe-practicas-preprofesionales-ing-sistemas-e-informatica3.shtml>.
22. Software Gaussian-09w. [En línea] <http://gaussian-09w.software.informer.com/&usg=ALkJrhhrXljP4AOB-ICNQLNzO1tayWUptw>.
23. Ecured: Visual Paradigm. [En línea] 2010. www.ecured.cu/index.php/Visual_Paradigm.
24. Rational Rose Enterprise . [En línea] <http://www.rational.com.ar/herramientas/roseenterprise.html>.
25. Lenguaje de Programación. [En línea] 2010. <http://es.scribd.com/doc/15270823/LENGUAJES-DE-PROGRAMACION-para-el-space>.
26. **Reynoso, Billy.***Profundizando en Estilos de Arquitectura de Software.* 2004.
27. *Material didáctico sobre la resolución de problemas en las asignaturas de Ciencias Exactas.* **Rodríguez Rodríguez, Dr. Luis E., Rodríguez Legrá, Dr. Donaciano y Sierralta Martínez, Prof. Lidia.**
28. **Myers, Glen.***The Art of Software Testing.* 1979. ISBN-10: 0471078786.
29. **Pressman, Roger S.***Ingeniería de Software. Un enfoque práctico.* s.l. : Mc Graw Hill, 2001.
30. Gadgetcabaret's Weblog. [En línea] 5 de diciembre de 2007.
<http://gadgetcabaret.wordpress.com/2007/12/05/netbeans-60-instalacion-y-caracteristicas/>.

31. Gadgetcabaret's Weblog. [En línea] 5 de diciembre de 2007.
<http://gadgetcabaret.wordpress.com/2007/12/05/netbeans-60-instalacion-y-caracteristicas/>.
32. **Pecos, Daniel.** PostGreSQL vs. MySQL. [En línea] http://danielpecos.com/docs/mysql_postgres/index.html.
33. **Dark, Malw.** Apuntes de análisis de sistemas. [En línea] <http://apuntessis.blogspot.com/2009/06/puds-analisis-iii.html>.
34. **Razumovski, V. G.** *Desarrollo de las capacidades creadoras de los estudiantes en el proceso de enseñanza de la Física*. La Habana : Pueblo y Educación. ., 1987. pág. pág. 2.

Bibliografía

Bibliografía.

1. Castañeda, E. (2003): *El papel de las tecnologías de la información y las comunicaciones (TIC) en el proceso de enseñanza-aprendizaje a comienzos del siglo XXI. Capítulo 10.* En Varela, F. (editor). *Preparación Pedagógica Integral para profesores universitarios.* Ciudad de La Habana: Félix Varela. p. 149
2. Castañeda, E. (2006): *Aplicaciones de las Nuevas Tecnologías de la Información y las Comunicaciones en el proceso de enseñanza aprendizaje.* En Conferencia No 2: "¿Qué modelo, que Gestor y qué Centro de Recursos Virtuales debo comprar?, ¿Cuándo y cómo debo hacerlo?".
3. Leyet, O. (2002): *Metodología para la utilización de las NTIC en el proceso de enseñanza-aprendizaje del idioma inglés en las carreras de ingeniería.* ISP "José Antonio Echeverría". Centro de Referencia para la Educación de Avanzada.
4. Rodríguez, J. (2003): *Una propuesta metodológica para la utilización de las TIC en el proceso de enseñanza aprendizaje de las funciones matemáticas.* ISPEJV.
5. Moreira, M. (2008): *¿Qué aporta Internet al cambio pedagógico en la Educación Superior?* publicado el: 2008 de 2006, última actualización: 2008.
6. Amaya, D. (2008): *Propuesta Metodológica para el perfeccionamiento del proceso de enseñanza de la Matemática Discreta en la UCI.* CREA. CUJAE.
7. Arcia, W. (2006): *Formación Matemática del Ingeniero Informático.* En *Formación Matemática del Ingeniero Informático.* Camagüey.
8. Delgado Cruz, Lic. Yeily (2011): *Estrategia Metodológica para desarrollar la interdisciplinariedad desde el PEA del Álgebra Lineal con la asignatura Gráficos por Computadoras a través de las TIC.* Capítulo 1. p. 149.
9. Ajona, Guillermo Prado. *Página Web sobre HTML y CSS.* *Página Web sobre HTML y CSS.* [En línea] <https://belenus.unirioja.es/~guprado/pagweb/caraccss.html>.

10. Jacobson, I. y Booch, G. y Rumbaugh. *El Proceso Unificado de Desarrollo de software*. s.l.: Addison-Wesley, 2000. ISBN: 84-7829-036-2.
11. John. *Manuales de ayuda.com*. *Manuales de ayuda.com*. [En línea] 23 de diciembre de 2006. <http://www.manualesdeayuda.com/manuales/bases-de-datos/postgresql/caracteristicas-de-postgresql-01844.html>.
12. Pressman, Roger S., *Ingeniería de Software. Un enfoque práctico*. s.l.: Mc Graw Hill, 2001.
13. Chaves, Michel Arias. *La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software*. Costa Rica: Revista InterSedes, 2007. Vol. VI, 10. ISSN 1409-4746.
14. Larman, Craig. *UML y Patrones: Introducción al análisis y diseño orientado a objetos*. México: Universitarios, 1999. ISBN: 970-17-0261-1.
15. Brito Acuña, Karennny. *Selección de Metodologías de Desarrollo para Aplicaciones Web en la Facultad de Informática de la Universidad de Cienfuegos*. [Tesis]. Cienfuegos, Cuba: s.n., 2009. Texto completo en www.eumed.net/libros/2009c/584/.
16. Letelier, Patricio y Penadés, M^a Carmen. *Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP)*. Universidad Politécnica de Valencia.
17. Montaldo, Diego Fernando. *Patrones de Diseño de Arquitecturas de Software Enterprise*. Tesis. Noviembre 2005. Departamento de Computación. Facultad de Ingeniería. Universidad de Buenos Aires
18. Wright, E.B. y Forcier, R.C. *The Computer: A Tool for the Teacher*. Wadsworth, Estados Unidos: s.n., 1985.

Anexos

Anexos.

Diagramas de clases del análisis.

Anexo 1: Diagrama de clases del análisis: CU_ Autenticar usuario.

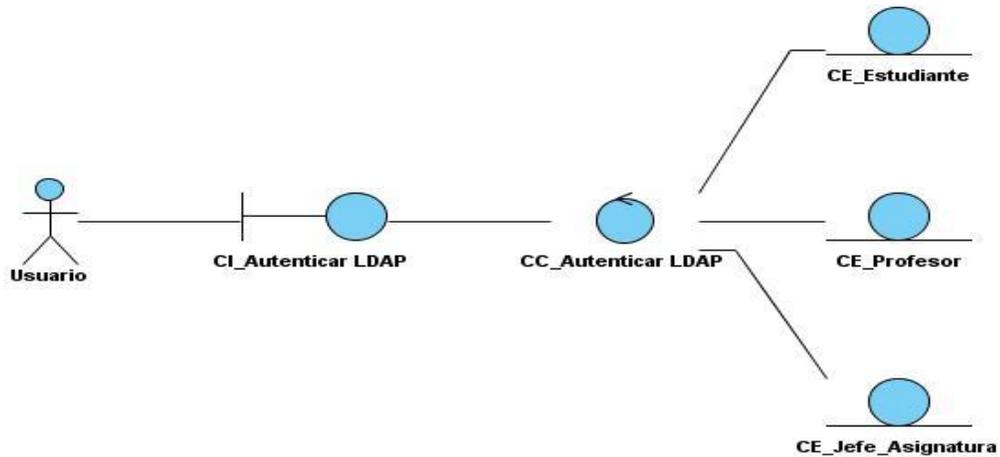


Ilustración 14: Diagrama de clases del análisis: CU_ Autenticar usuario

Anexo 2: Diagrama de clases del análisis: CU_ Gestionar Profesores.

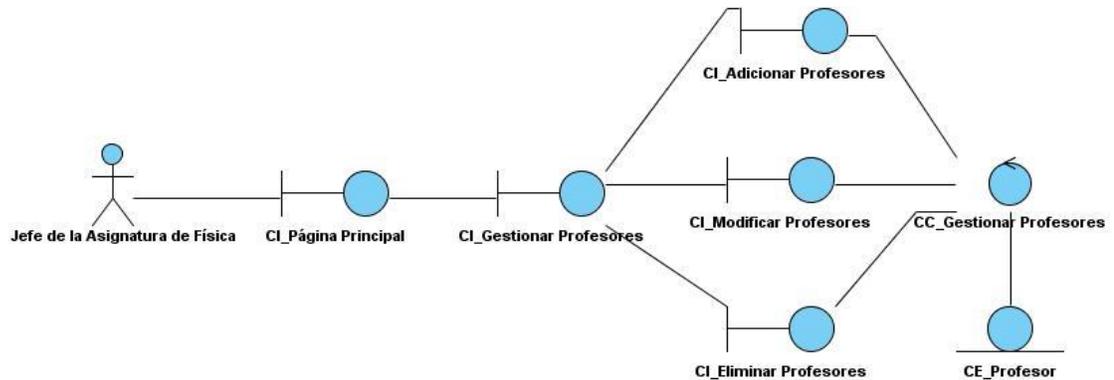


Ilustración 15: Diagrama de clases del análisis: CU_ Gestionar Profesores

Anexo 3: Diagrama de clases del análisis: CU_ Gestionar Grupo

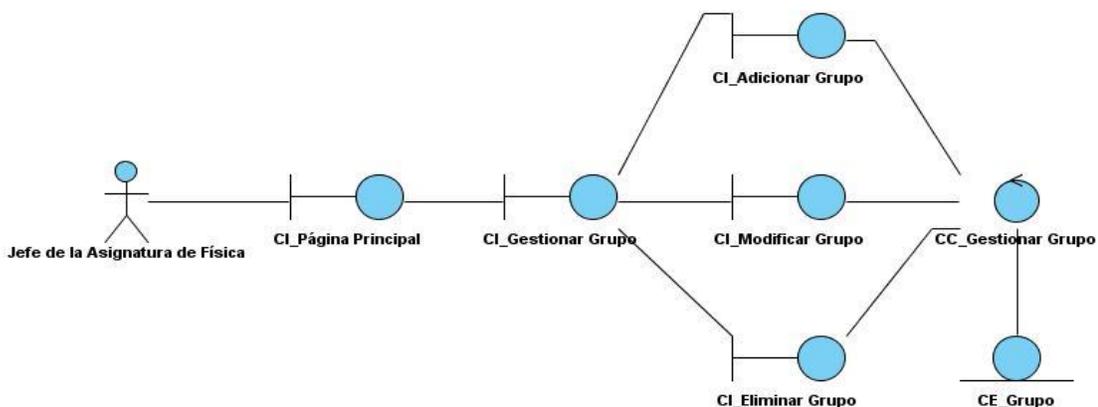


Ilustración 16: Diagrama de clases del análisis: CU_ Gestionar Grupo

Anexo 4: Diagrama de clases del análisis: CU_ Asignar Ejercicio.

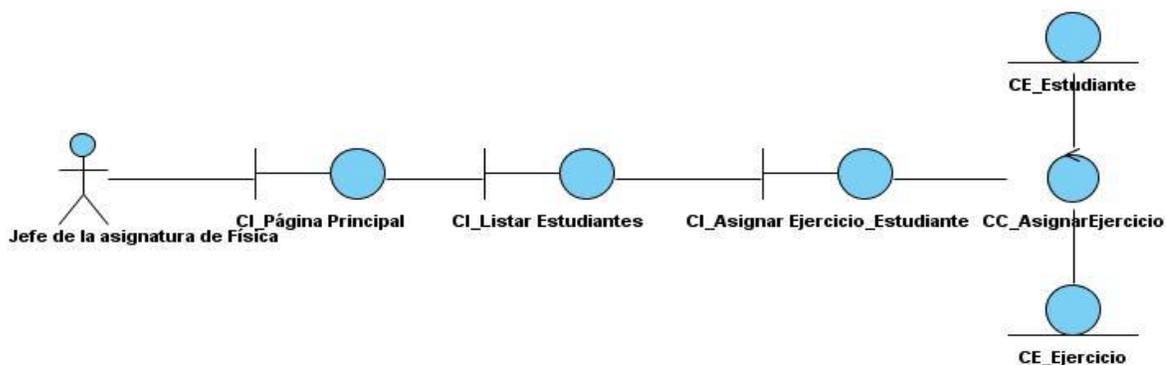


Ilustración 17: Diagrama de clases del análisis: CU_ Asignar Ejercicio

Anexo 5: Diagrama de clases del análisis: CU_ Generar Reporte

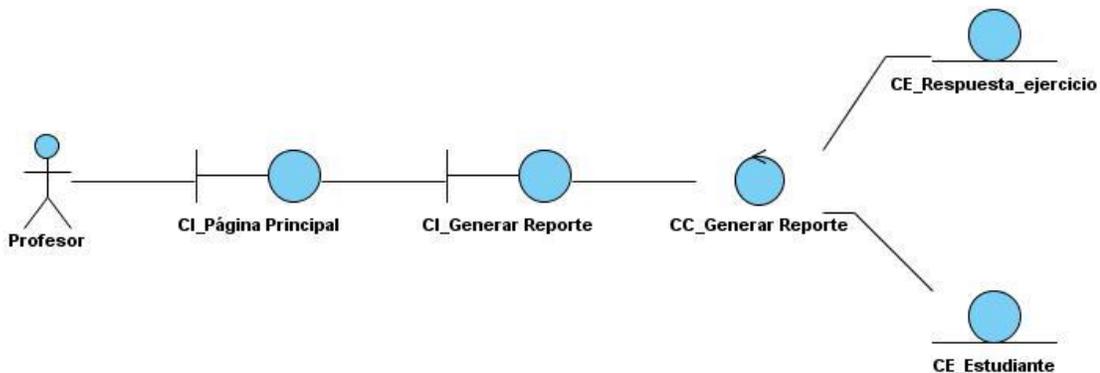


Ilustración 18: Diagrama de clases del análisis: CU_ Generar Reporte

Diagramas de Interacción (Colaboración).

Anexo 6: Diagrama de colaboración: CU_ Autenticar Usuario.

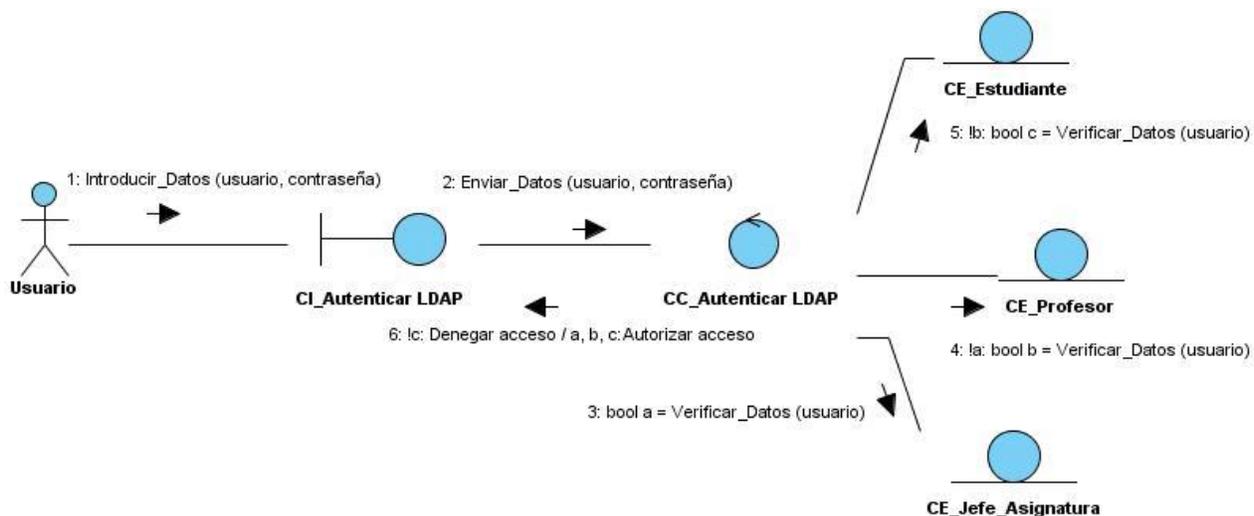


Ilustración 19: Diagrama de clases del análisis: CU_ Autenticar Usuario

Anexo 7: Diagrama de colaboración: CU_ Gestionar Profesores.

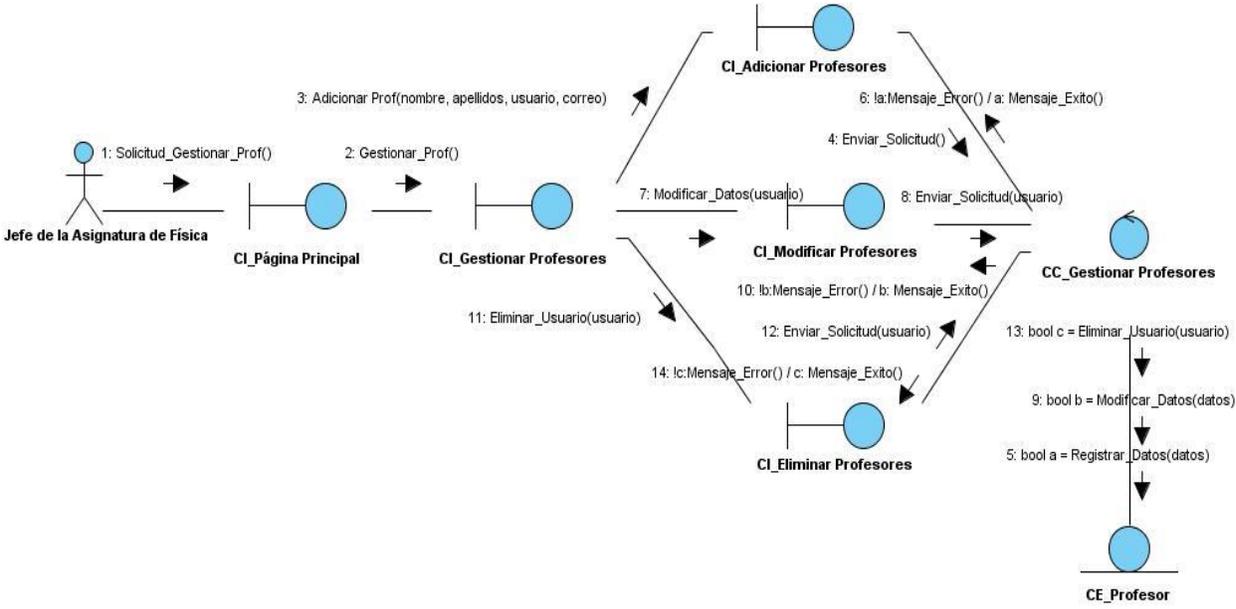


Ilustración 20: Diagrama de clases del análisis: CU_ Gestionar Profesores

Anexo 8: Diagrama de colaboración: CU_ Gestionar Grupo.

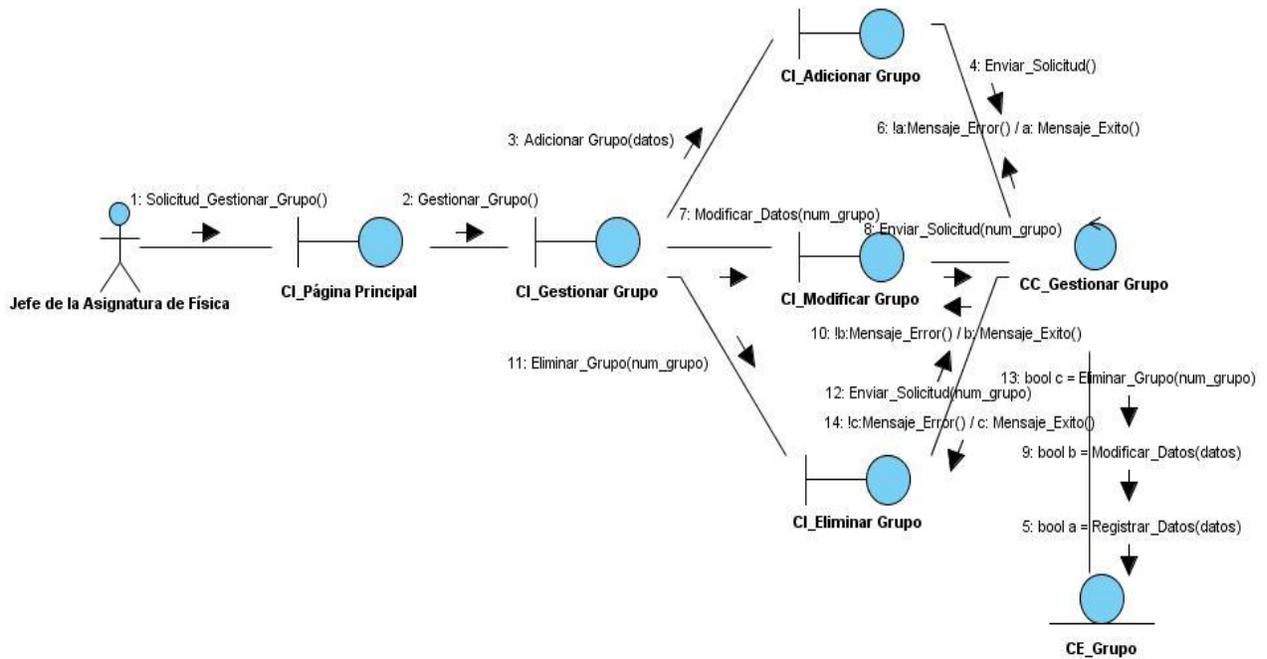


Ilustración 21: Diagrama de clases del análisis: CU_ Gestionar Grupo

Anexo 9: Diagrama de colaboración: CU_ Asignar Ejercicios.

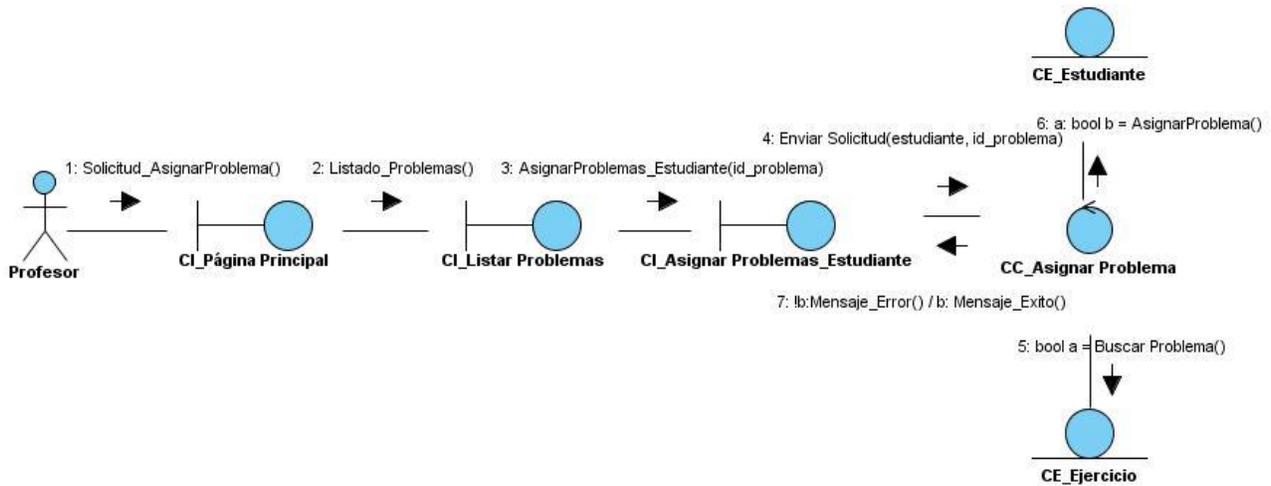


Ilustración 22: Diagrama de clases del análisis: CU_ Asignar Ejercicios.

Anexo 10: Diagrama de colaboración: CU_ Generar Reporte.

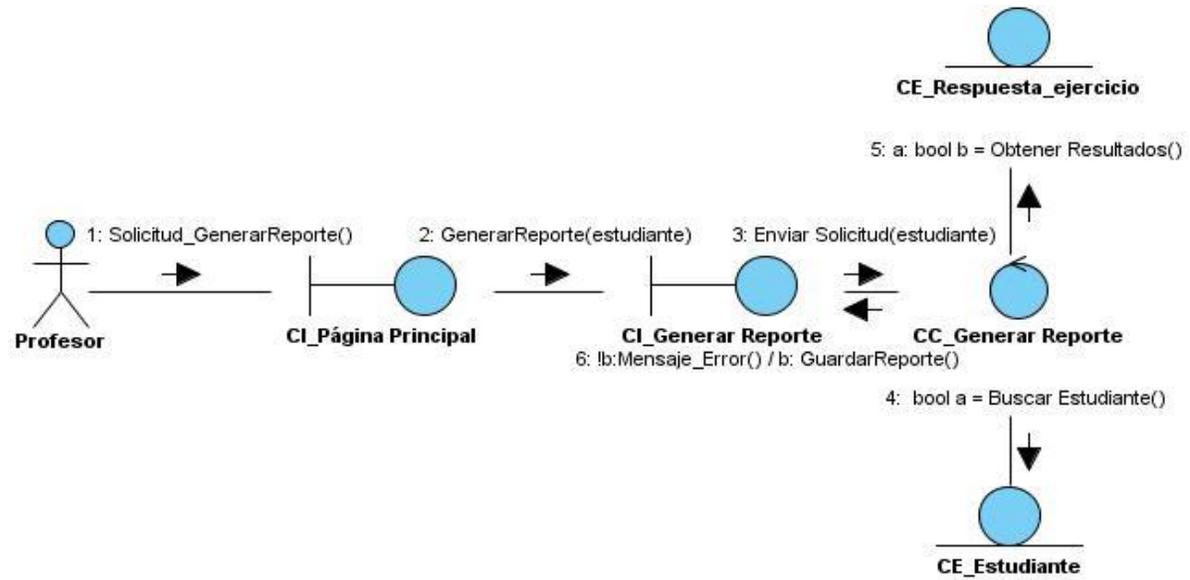


Ilustración 23: Diagrama de clases del análisis: CU_ Generar Reporte

Diagramas de clases del diseño con extensiones web.

Anexo 11: Diagrama de clases del diseño: CU_ Generar Reporte.

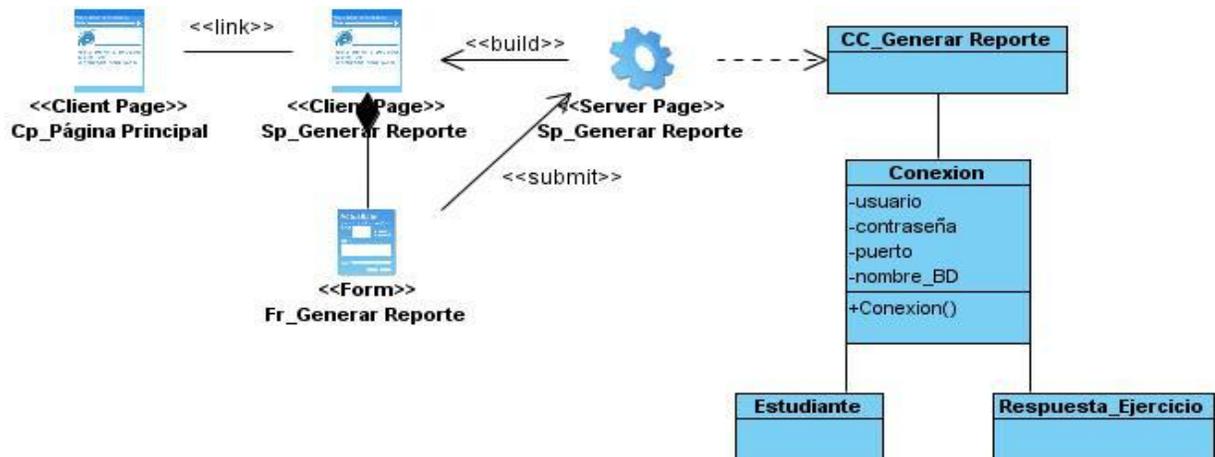


Ilustración 24: Diagrama de clases del diseño: CU_ Generar Reporte

Anexo 12: Diagrama de clases del diseño: CU_Gestionar Profesores.

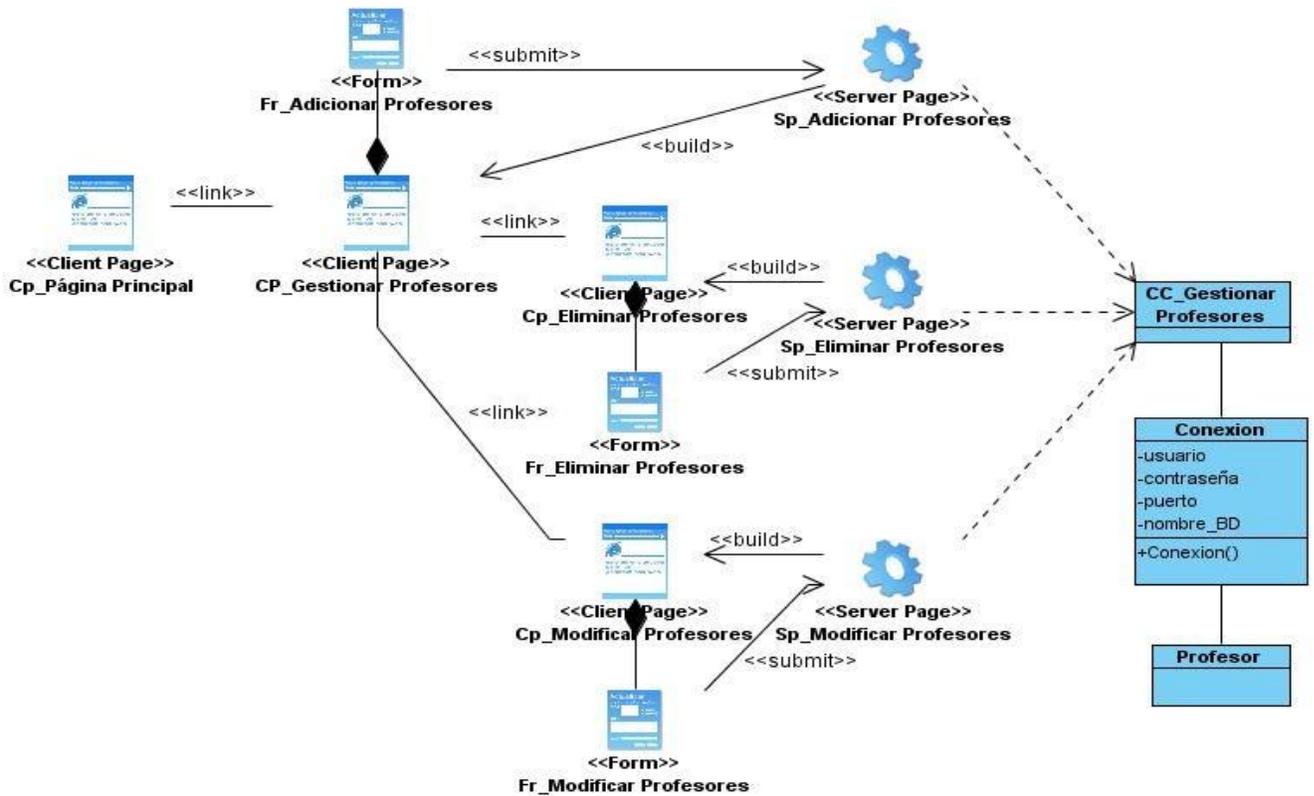


Ilustración 25: Diagrama de clases del diseño: CU_Gestionar Profesores

Anexo 13: Diagrama de clases del diseño: CU_Asignar Problema.

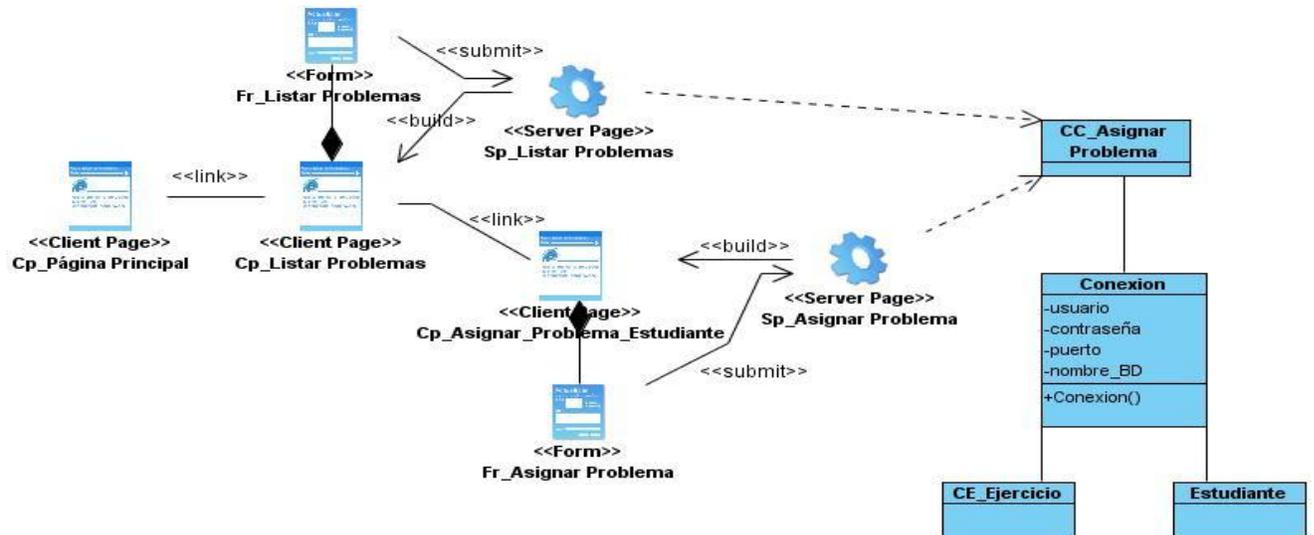


Ilustración 26: Diagrama de clases del diseño: CU_Asignar Problema

Anexo 14: Diagrama de clases del diseño: CU_ Autenticar Usuario.

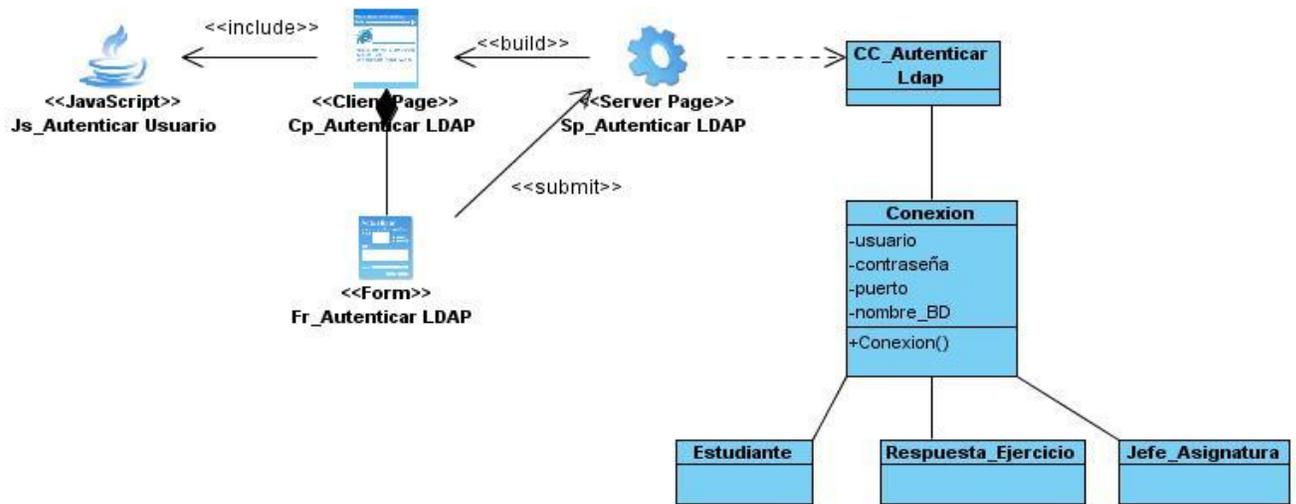


Ilustración 27: Diagrama de clases del diseño: CU_ Autenticar Usuario

Anexo 15: Diagrama de clases del diseño: CU_ Gestionar Grupo.

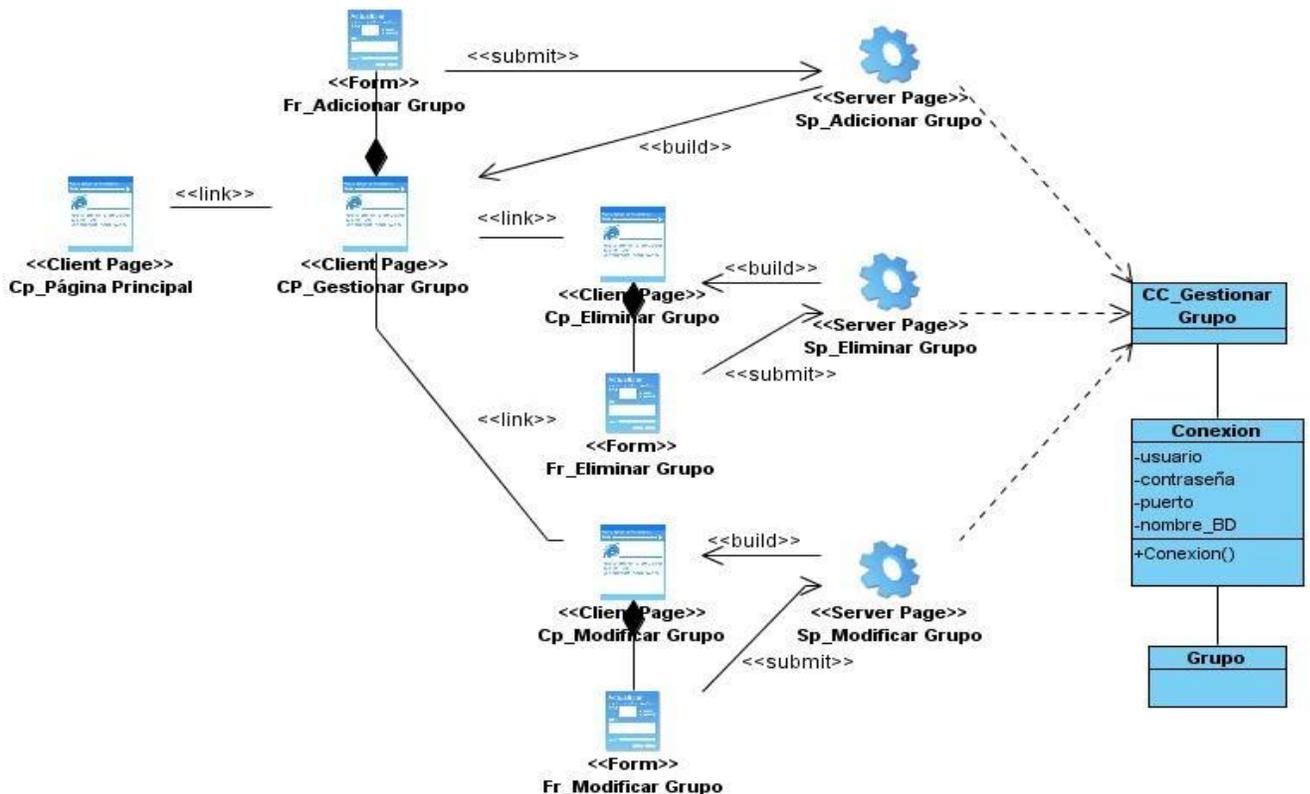


Ilustración 28: Diagrama de clases del diseño: CU_ Gestionar Grupo

Diseño de Casos de Prueba.

Anexo 15: Caso de Prueba: Gestionar profesores.

1: Nombre del profesor.

2: Primer apellido.

3: Segundo apellido.

4: Usuario UCI.

5: Correo UCI.

| Escenario de la sección | 1 | 2 | 3 | 4 | 5 | Respuesta del sistema | Flujo central |
|---|-----|-----|-----|-----|-----|---|--|
| EC3.1 Alguno de los campos del formulario en blanco. | I | I | I | I | I | Muestra un mensaje de error indicando: "No se permiten campos vacíos". | <ol style="list-style-type: none"> Al introducir los datos para el nuevo profesor se dejan algunos de los campos requeridos vacíos. Se presiona el botón Enviar. |
| EC3.2 Profesor registrado con éxito. | V | V | V | V | V | Muestra un mensaje de éxito de la operación indicando: "Profesor registrado correctamente". | <ol style="list-style-type: none"> Se introducen los datos correctamente. Se presiona el botón Enviar. Se muestra la notificación "Profesor registrado correctamente". |
| EC 2.3 Números o Caracteres extraños en el campo del nombre del profesor | I | N/A | N/A | N/A | N/A | Muestra un mensaje de error indicando: "Nombre del profesor no permitido". | <ol style="list-style-type: none"> Se introducen los datos incorrectos para el Nombre del profesor. Se presiona el botón Enviar. Se muestra la notificación de error indicando: "Nombre del profesor no permitido". |
| EC 2.4 Números o Caracteres extraños en el campo Primer | N/A | I | N/A | N/A | N/A | Muestra un mensaje de error indicando: "Primer apellido no permitido". | <ol style="list-style-type: none"> Se introducen los datos incorrectos para el primer apellido del profesor. Se presiona el botón Enviar. Se muestra la notificación de error indicando: "Primer |

| | | | | | | | |
|--|-----|-----|-----|-----|-----|--|---|
| apellido | | | | | | | apellido no permitido”. |
| EC 2.5 Números o Caracteres extraños en el campo Segundo apellido | N/A | N/A | I | N/A | N/A | Muestra un mensaje de error indicando: “Segundo apellido no permitido”. | <ol style="list-style-type: none"> 1. Se introducen los datos incorrectos para el segundo apellido del profesor. 2. Se presiona el botón Enviar. 3. Se muestra la notificación de error indicando: “Segundo apellido no permitido”. |
| EC 2.6 Números o Caracteres extraños en el campo Usuario UCI. | N/A | N/A | N/A | I | N/A | Muestra un mensaje de error indicando: “Usuario no permitido”. | <ol style="list-style-type: none"> 1. Se introducen los datos incorrectos para el usuario del profesor. 2. Se presiona el botón Enviar. 3. Se muestra la notificación de error indicando: “Usuario no permitido”. |
| EC 2.7 Números, Caracteres extraños o un formato fuera de @uci.cu en el campo Correo UCI. | N/A | N/A | N/A | N/A | I | Muestra un mensaje de error indicando: “Correo no permitido”. | <ol style="list-style-type: none"> 1. Se introducen los datos incorrectos para el correo del profesor. 2. Se presiona el botón Enviar. 3. Se muestra la notificación de error indicando: “Correo no permitido”. |
| EC 2.8 Diferentes los datos para el usuario registrado con el del campo Correo UCI. | N/A | N/A | N/A | N/A | I | Muestra un mensaje de error indicando: “Verifique el usuario con el correo”. | <ol style="list-style-type: none"> 1. Se introducen los datos para el usuario diferentes al del usuario del correo del profesor. 2. Se presiona el botón Enviar. 3. Se muestra la notificación de error indicando: “Verifique el usuario con el correo”. |

Anexo 16: Caso de Prueba: Gestionar grupo.

1: Número del grupo.

2: Profesor.

| Escenario de la sección | 1 | 2 | Respuesta del sistema | Flujo central |
|---|-----|-----|--|--|
| EC 4.1 Alguno de los campos del formulario en blanco. | I | I | Muestra un mensaje de error indicando: "No se permiten campos vacíos". | <ol style="list-style-type: none"> 1. Al introducir los datos para el nuevo grupo se deja alguno de los campos requeridos vacíos. 2. Se presiona el botón Enviar. |
| EC 4.2 Grupo registrado con éxito. | V | V | Muestra un mensaje de éxito de la operación indicando: "Grupo registrado correctamente". | <ol style="list-style-type: none"> 1. Se introducen los datos correctamente. 2. Se presiona el botón Enviar. 3. Se muestra la notificación "Grupo registrado correctamente". |
| EC 4.3 Caracteres extraños en el campo del número del grupo | I | N/A | Muestra un mensaje de error indicando: "Número del grupo no permitido". | <ol style="list-style-type: none"> 1. Se introducen los datos incorrectos para el Nombre del profesor. 2. Se presiona el botón Enviar. 3. Se muestra la notificación de error indicando: "Número del grupo no permitido". |
| EC 4.4 No se selecciona el profesor para que atienda al grupo creado. | N/A | I | Muestra un mensaje de error indicando: "Debes seleccionar un profesor". | <ol style="list-style-type: none"> 1. No se selecciona el profesor que atenderá el nuevo grupo. 2. Se presiona el botón Enviar. 3. Se muestra la notificación de error indicando: "Debes seleccionar un profesor". |
| EC 4.5 No se indica un número correcto para el grupo (<i>debe empezar por 52</i>). | I | N/A | Muestra un mensaje de error indicando: "Número del grupo no permitido". | <ol style="list-style-type: none"> 1. Se introducen los datos incorrectos para el Nombre del profesor. 2. Se presiona el botón Enviar. 3. Se muestra la notificación de error indicando: "Número del grupo no permitido". |

Anexo 17: Caso de Prueba: Gestionar respuestas.

- 1: Selección de la región a analizar.
- 2: Plantear la ley de Gauss.
- 3: Determinar el valor numérico del ángulo que forma la intensidad del campo eléctrico con el diferencial de superficie.
- 4: Despejar la intensidad del campo eléctrico y calcular la integral del diferencial de superficie.
- 5: Determinar la expresión para el cálculo de la intensidad de campo eléctrico en la región indicada.
- 6: Indicar la unidad de medida.

| Escenario de la sección | 1 | 2 | 3 | 4 | 5 | 6 | Respuesta del sistema | Flujo central |
|--|-----|-----|-----|-----|-----|-----|--|--|
| EC 5.1 Deja al menos un campo en blanco | I | I | I | I | I | I | Muestra un mensaje de error indicando: "Debe llenar todos los campos". | <ol style="list-style-type: none"> 1. Al introducir los datos para la resolución del ejercicio se deja algún campo vacío. 2. Se presiona el botón Enviar. |
| EC 5.2 Resolución correcta del ejercicio | V | V | V | V | V | V | Muestra un mensaje de éxito de la operación indicando que el ejercicio ha sido resuelto y muestra además la evaluación en el respectivo campo: "Ejercicio resuelto con éxito". | <ol style="list-style-type: none"> 1. Se introducen los datos en todos los campos. 2. Se presiona el botón Enviar. 3. Se muestra la notificación y la evaluación del ejercicio. "Ejercicio resuelto con éxito". |
| EC 5.3 No selecciona la región a analizar | N/A | N/A | N/A | N/A | N/A | N/A | Muestra un mensaje de error indicando: "Debe seleccionar la región analizar". | <ol style="list-style-type: none"> 1. No se selecciona la región a analizar. 2. Se llenan todos los campos y se presiona el botón Enviar. |

Anexo 18:

Tabla 1: Especificación de caso de uso: Autenticar usuario

| | | |
|--|---|---|
| Caso de Uso: | CU_Autenticar usuario. | |
| Objetivo | Permitir el acceso inicial a la aplicación. | |
| Actores | Usuario. | |
| Resumen | El caso de uso inicia cuando el actor introduce los datos requeridos para entrar en el sistema, este caso de uso es el encargado de proporcionar el acceso inicial a la aplicación a solo aquellos usuarios autorizados, usando el servicio ldap que brinda la universidad. | |
| Complejidad | Baja. | |
| Requisitos de Referencia | RF 1 | |
| Prioridad | Crítico. | |
| Precondiciones | El usuario debe pertenecer al dominio UCI. | |
| Flujo de eventos | | |
| Flujo básico "Autenticar usuario" | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Introduce los datos en los campo requeridos para el acceso inicial y presiona el botón Enviar. (Alternativo 1, Alternativo 2, Alternativo 3). | 1.1 Verifica que el campo de usuario no contenga números ni caracteres extraños. 1.2 Verifica que los datos entrados correspondan a estudiantes de segundo año o profesores de Física de la facultad 5. 1.3 Muestra la página inicial de cada usuario en dependencia del rol identificado. 1.4 Termina el caso de uso. |
| Flujos alternos | | |

| No | Acción del Actor | Respuesta del Sistema |
|----|--|--|
| 1. | En el paso 1 del flujo básico de eventos introduce números o caracteres extraños en el campo Usuario y presiona el botón Enviar. | <p>1.1 Retorna a la página de acceso brindando nuevamente la posibilidad de entrar los datos, mostrando un mensaje de error indicando: <i>“Error en el campo Usuario”</i>.</p> <p>1.2 Limpia los campos de usuario y contraseña.</p> |
| 2. | Deja campos en blanco y presiona el botón Enviar. | <p>2.1 Retorna a la página de acceso brindando nuevamente la posibilidad de entrar los datos.</p> <p>2.2 Muestra un mensaje de error indicando: <i>“Debe llenar todos los campos”</i>.</p> <p>2.3 Limpia los campos de usuario y contraseña.</p> |
| 3. | Introduce usuario o contraseña distintas a las del dominio uci y presiona el botón Enviar. | <p>3.1 Retorna a la página de acceso brindando nuevamente la posibilidad de entrar los datos.</p> <p>3.2 Muestra un mensaje de error indicando: <i>“Usuario o Contraseña inválido”</i>.</p> <p>3.3 El sistema limpia los campos de usuario y contraseña.</p> |

Prototipo de interfaz



| | |
|-----------------------|---|
| Poscondiciones | Se le muestra al usuario la interfaz correspondiente al rol autenticado, así como sus funcionalidades y los permisos recibidos. |
|-----------------------|---|

Anexo 18:

Tabla 2 Especificación de caso de uso: Gestionar profesores

| | | |
|---|--|---|
| Caso de Uso: | CU_ Gestionar Profesores. | |
| Objetivo | Gestionar cada uno de los profesores de Física de la facultad. | |
| Actores | Jefe de la asignatura de Física. | |
| Resumen | El caso de uso inicia cuando el Jefe de la asignatura de Física decide adicionar, modificar o eliminar datos acerca de un profesor que posteriormente interactuará con la aplicación. | |
| Complejidad | Media. | |
| Requisitos de Referencia | RF 3.1, 3.2, 3.3 | |
| Prioridad | Crítico. | |
| Precondiciones | El actor debe estar autenticado como Jefe de la Asignatura de Física y acceder a la opción de Gestionar Profesores. El jefe de la asignatura antes de eliminar a un profesor del grupo docente que atiende tendrá que asignárselo a otro profesor previamente registrado. | |
| Flujo de eventos | | |
| Flujo básico “Escenario adicionar” | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Selecciona la opción Gestionar Profesores del menú principal. | 1.1 Muestra el formulario Adicionar Profesor con los siguientes campos: <ul style="list-style-type: none"> • Nombre • Primer Apellido |

| | | |
|--|---|---|
| | | <ul style="list-style-type: none"> • Segundo Apellido • Usuario UCI • Correo UCI <p>1.2 Muestra el botón Profesores que permite listar los que se encuentran registrados para una posterior modificación u/o eliminación.</p> |
| 2. | Llena los campos requeridos para que el nuevo profesor y presiona Enviar. (Alternativo 1, Alternativo 2, Alternativo 3, Alternativo 4, Alternativo 5, Alternativo 6 y Alternativo 7). | <p>2.1 Verifica que todos los campos estén completados.</p> <p>2.2 Verifica que todos los campos llenados no contiene caracteres extraños.</p> <p>2.3 Verifica que el usuario se corresponda con el registrado en el correo.</p> <p>2.4 Verifica que el formato del correo sea válido para un profesor de la universidad.</p> <p>2.5 Verifica en la BD que el usuario no existe.</p> <p>2.6 Retorna al formulario Adicionar nuevo profesor mostrando un mensaje de éxito de la operación: "Profesor registrado correctamente".</p> <p>2.7 Termina el caso de uso.</p> |
| Flujos alternos (Escenario adicional) | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | En el paso 2 del flujo normal de eventos introduce números o caracteres extraños en el campo Nombre y presiona el botón Enviar. | <p>1.1 Retorna a la página de gestionar profesores brindando nuevamente la posibilidad de entrar los datos.</p> <p>1.2 Muestra un mensaje de error</p> |

| | | |
|----|--|--|
| | | <p>indicando: <i>"Nombre del profesor no permitido"</i>.</p> <p>1.3 Limpia los campos para la nueva entrada de los datos.</p> |
| 2. | <p>En el paso 2 del flujo normal de eventos introduce números o caracteres extraños en el campo Primer apellido y presiona el botón Enviar.</p> | <p>2.1 Retorna a la página de gestionar profesores brindando nuevamente la posibilidad de entrar los datos.</p> <p>2.2 Muestra un mensaje de error indicando: <i>"Primer apellido no permitido"</i>.</p> <p>2.3 Limpia los campos para la nueva entrada de los datos.</p> |
| 3. | <p>En el paso 2 del flujo normal de eventos introduce números o caracteres extraños en el campo Segundo apellido y presiona el botón Enviar.</p> | <p>3.1 Retorna a la página de gestionar profesores brindando nuevamente la posibilidad de entrar los datos.</p> <p>3.2 Muestra un mensaje de error indicando: <i>"Segundo apellido no permitido"</i>.</p> <p>3.3 Limpia los campos para la nueva entrada de los datos.</p> |
| 4. | <p>En el paso 2 del flujo normal de eventos introduce números o caracteres extraños en el campo Usuario UCI y presiona el botón Enviar.</p> | <p>4.1 Retorna a la página de gestionar profesores brindando nuevamente la posibilidad de entrar los datos.</p> <p>4.2 Muestra un mensaje de error indicando: <i>"Usuario no permitido"</i>.</p> <p>4.3 Limpia los campos para la nueva entrada de los datos.</p> |
| 5. | <p>En el paso 2 del flujo normal de eventos introduce números, caracteres extraños o un formato fuera del @uci.cu en el campo Correo UCI y presiona el botón Enviar.</p> | <p>5.1 Retorna a la página de gestionar profesores brindando nuevamente la posibilidad de entrar los datos.</p> <p>5.2 Muestra un mensaje de error indicando: <i>"Correo no permitido"</i>.</p> |

| | | |
|----|---|---|
| | | 5.3 Limpia los campos para la nueva entrada de los datos. |
| 6. | En el paso 2 del flujo normal de eventos introduce el Usuario UCI y el usuario del Correo UCI diferente y presiona el botón Enviar. | 6.1 Retorna a la página de gestionar profesores brindando nuevamente la posibilidad de entrar los datos. 6.2 Muestra un mensaje de error indicando: <i>“Verifique el usuario con el correo”</i> . 6.3 Limpia los campos para la nueva entrada de los datos. |
| 7. | En el paso 2 del flujo normal de eventos presiona el botón Enviar teniendo aun campos vacíos. | 7.1 Muestra un mensaje de error indicando: <i>“No se permiten campos vacíos”</i> . 7.2 Retorna al formulario principal. |

Prototipo de interfaz

Flujo de eventos

Flujo básico “ Escenario modificar”

| No | Acción del Actor | Respuesta del Sistema |
|----|------------------|-----------------------|
|----|------------------|-----------------------|

| | | |
|----|---|--|
| 1. | Presiona el botón Profesores en el escenario Adicionar nuevo Profesor. | 1.1 Muestra una interfaz donde son listados todos los profesores de la asignatura de Física en la facultad que pueden ser modificados. |
| 2. | Presiona el botón  presente en el lateral derecho del profesor que desea modificar sus datos. | 2.1 Muestra la interfaz Modificar Profesor con los valores existentes en la base de datos en cada uno de los campos. |
| 3. | Modifica los datos pertinentes y presiona el botón Enviar. (Alternativo 1, Alternativo 2, Alternativo 3, Alternativo 4, Alternativo 5, Alternativo 6 y Alternativo 7 del Escenario Adicionar Profesor). | <p>3.1 Verifica que todos los campos estén llenos.</p> <p>3.2 Verifica que todos los campos llenados no contienen caracteres extraños.</p> <p>3.3 Verifica que el formato del correo sea válido para un profesor de la universidad.</p> <p>3.4 Verifica en la BD que el usuario no existe (<i>en caso de ser modificado ese campo</i>).</p> <p>3.5 Retorna al formulario Modificar Profesor mostrando un mensaje de éxito de la operación: " <i>Profesor modificado correctamente</i>".</p> <p>3.6 Termina el caso de uso.</p> |

Prototipo de interfaz

| Nombre | 1er Apellido | 2do Apellido | Modificar | Eliminar |
|---------------|--------------|--------------|---|---|
| Yidian Yosbel | Castellanos | Sabari |  |  |
| Yoandys | Hernandez | Alvarez |  |  |
| Brenda | Batista | Fons |  |  |
| Yalena | Venega | Canizares |  |  |

Flujo de eventos

| Flujo básico “ Escenario eliminar ” | | |
|--------------------------------------|---|--|
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Presiona el botón Profesores en el escenario Adicionar nuevo Profesor. | 1.1 Muestra una interfaz donde son listados todos los profesores de la asignatura de Física en la facultad que pueden ser eliminados. |
| 2. | Presiona el botón  presente en el lateral derecho del profesor que se desea eliminar. (Alternativo 1) | 2.1 Verifica que el profesor no tiene grupos asignados. 2.2 Elimina todos los datos del profesor seleccionado. 2.3 Retorna a la interfaz del listado de profesores registrados. 2.4 Termina el caso de uso. |
| Flujos alternos (Escenario eliminar) | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | En el paso 2 del flujo normal de eventos presiona el botón  presente en el lateral derecho del profesor que se desea eliminar teniendo grupos asignados. | 1.1 Muestra el listado de los profesores y sus respectivos grupos asignados, brindando la posibilidad de asignar su grupo a otro profesor. |
| 2. | Modifica el grupo que atiende ese profesor asignándose a otro y acude al escenario Eliminar profesor. | 2.1 Elimina todos los datos del profesor seleccionado. 2.2 Retorna a la interfaz del listado de profesores registrados. |
| Poscondiciones | | Se le muestra al usuario cada una de las interfaces de adicionar, modificar o eliminar un profesor según sea el caso. |

Anexo 19:

Tabla 3 Especificación de caso de uso: Gestionar Grupo

| Caso de Uso: | CU_ Gestionar Grupo | |
|---|---|---|
| Objetivo | Gestionar cada uno de los grupos que reciben la asignatura de Física de la facultad. | |
| Actores | Jefe de la asignatura de Física. | |
| Resumen | El caso de uso inicia cuando el Jefe de la asignatura de Física lleva a cabo todo el proceso de gestión de los distintos grupos en la facultad que reciben la asignatura de Física ya sea adicionar, eliminar y/o modificar uno de ellos. | |
| Complejidad | Media. | |
| Requisitos de Referencia | RF 4.1, 4.2, 4.3 | |
| Prioridad | Crítico. | |
| Precondiciones | El actor debe estar autenticado previamente como Jefe de la asignatura. | |
| Flujo de eventos | | |
| Flujo básico "Escenario adicionar" | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Selecciona la opción Gestionar Grupos del menú principal. | <p>1.1 Muestra el formulario Adicionar nuevos grupos con los siguientes campos:</p> <ul style="list-style-type: none"> • Número del Grupo • Profesor <p>1.2 Muestra además el botón Grupos que muestra el listado de todos los grupos registrados con sus respectivos profesores.</p> |
| 2. | Llena el campo requerido para el | 2.1 Verifica que todos los campos estén |

| | número del grupo, selecciona el profesor que lo atenderá y presiona Enviar. (Alternativo 1, Alternativo 2 y Alternativo 3). | <p>lentos, en el caso de haber algún campo vacío lo re direcciona nuevamente al formulario mostrando un mensaje de error al usuario que el campo que falta es obligatorio.</p> <p>2.2 Verifica que el campo Número del Grupo no contiene ni letras ni caracteres extraños.</p> <p>2.3 Verifica que el número del grupo comienza por 52.</p> <p>2.4 Verifica en la BD que el grupo no existe.</p> <p>2.5 Verifica que se seleccione un profesor para que atienda al grupo.</p> <p>2.6 Retorna al formulario Adicionar nuevo grupo mostrando un mensaje de éxito de la operación: <i>"Grupo registrado correctamente"</i>.</p> <p>2.7 Termina el caso de uso.</p> |
|--|--|---|
| Flujos alternos (Escenario adicional) | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | En el paso 2 del flujo normal de eventos introduce letras, caracteres extraños en el campo número del grupo o alguna combinación numérica que no comience con 52 y presiona el botón Enviar. | <p>1.1 Retorna a la página de gestionar grupo brindando nuevamente la posibilidad de entrar los datos.</p> <p>1.2 Muestra un mensaje de error indicando: <i>"Número del grupo no permitido"</i>.</p> <p>1.3 Limpia los campos para la nueva entrada de los datos.</p> |
| 2. | En el paso 2 del flujo normal de eventos presiona el botón Enviar teniendo aun | 2.1 Muestra un mensaje de error indicando: <i>"No se permiten campos vacíos"</i> . |

| | | |
|----|--|---|
| | campos vacíos. | 2.2 Retorna al formulario principal. |
| 3. | En el paso 2 del flujo normal de eventos no selecciona el profesor que atenderá el grupo y presiona el botón Enviar. | 3.1 Retorna a la página de gestionar grupo brindando nuevamente la posibilidad de entrar los datos. 3.2 Muestra un mensaje de error indicando: <i>“Debes seleccionar un profesor”</i> . 3.3 Limpia los campos para la nueva entrada de los datos. |

Prototipo de interfaz

Flujo de eventos

Flujo básico “ Escenario modificar ”

| No | Acción del Actor | Respuesta del Sistema |
|----|---|---|
| 1. | Presiona el botón Grupos en el escenario Adicionar nuevo grupo. | 1.1 Muestra una interfaz donde son listados todos los grupos con sus respectivos profesores que pueden ser modificados. |
| 2. | Presiona el botón  presente en el lateral derecho del grupo que desea modificar sus datos. | 2.1 Muestra la interfaz Modificar Grupo con los valores existentes en la base de datos en cada uno de los campos. |
| 3. | Modifica los datos pertinentes y | 3.1 Verifica que todos los campos estén |

| | | |
|--|--|--|
| | <p>presiona el botón Enviar. (Alternativo 1, Alternativo 2 y Alternativo 3 del Escenario Adicionar nuevo grupo).</p> | <p>lentos.</p> <p>3.2 Verifica que el campo Número del grupo no contiene letras ni caracteres extraños ni ninguna combinación numérica inicial fuera de 52.</p> <p>3.3 Verifica en la BD que el número del grupo no existe (<i>en caso de ser modificado ese campo</i>).</p> <p>3.4 Retorna al formulario Modificar Grupo mostrando un mensaje de éxito de la operación: " Grupo modificado correctamente".</p> <p>3.5 Termina el caso de uso.</p> |
|--|--|--|

Prototipo de interfaz

| Número del Grupo | Usuario del profesor | Modificar | Eliminar |
|------------------|----------------------|-----------|----------|
| 5203 | yhalvarez | ✓ | ✗ |
| 5207 | yycastellanos | ✓ | ✗ |
| 5201 | yycastellanos | ✓ | ✗ |
| 5204 | yhalvarez | ✓ | ✗ |
| 5202 | yhalvarez | ✓ | ✗ |

Flujo de eventos

Flujo básico " Escenario eliminar "

| No | Acción del Actor | Respuesta del Sistema |
|----|---|---|
| 1. | Presiona el botón Grupos en el escenario Adicionar nuevo grupo. | 1.1 Muestra una interfaz donde son listados todos los grupos con sus respectivos profesores que pueden ser eliminados. |
| 2. | Presiona el botón ✗ presente en el lateral derecho del grupo que se desea eliminar. | 2.1 Elimina todos los datos de la base de datos del grupo seleccionado. 2.2 Retorna a la interfaz del listado de grupos registrados. |

| | |
|-----------------------|---|
| | 2.3 Termina el caso de uso. |
| Poscondiciones | Se le muestra al usuario cada una de las interfaces de adicionar, modificar o eliminar un grupo según la selección realizada. |

Anexo 20:

Tabla 5 Especificación de caso de uso: Gestionar Respuestas

| | | |
|---|--|--|
| Caso de Uso: | CU_ Gestionar respuestas. | |
| Objetivo | Gestionar cada una de las respuestas de los ejercicios para posterior evaluación de los estudiantes. | |
| Actores | Jefe de la asignatura de Física. | |
| Resumen | El caso de uso inicia cuando el Jefe de la asignatura de Física lleva a cabo todo el proceso de gestión de las respuestas de un ejercicio seleccionado, de las cuales el estudiante podrá comprobar las registradas por él, además de poder eliminar una respuesta de una región seleccionada. | |
| Complejidad | Alta. | |
| Requisitos de Referencia | RF 6.1, 6.2 | |
| Prioridad | Crítico. | |
| Precondiciones | El actor debe estar autenticado como Jefe de la asignatura de Física y acceder a la opción de Gestionar respuestas. | |
| Flujo de eventos | | |
| Flujo básico “Escenario registrar” | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Selecciona la opción Gestionar Respuestas del menú principal. | 1.1 Muestra el formulario con el listado de los ejercicios registrados y las opciones Resolver y Eliminar Respuesta. |
| 2. | Selecciona la opción Resolver dando clic en la siguiente imagen  . | 2.1 Muestra el formulario Resolver ejercicio con cada uno de los campos |

| | | <p>que le deberá proporcionar una respuesta, como son:</p> <ul style="list-style-type: none"> • Seleccionar el área a analizar. • Plantear la ley de Gauss. • Determinar el ángulo que forma la intensidad del campo eléctrico con el diferencial de superficie. • Valor de la carga neta • Despeje y cálculo de la integral del diferencial de superficie. • Determinar la expresión para el cálculo de la intensidad de campo eléctrico. • Indicar la unidad de medida. |
|--|--|--|
| 3. | Registra las respuestas indicadas para cada uno de los campos y presiona Enviar. (Alternativo 1, Alternativo 2 y Alternativo 3). | <p>3.1 Verifica que no exista ningún campo vacío.</p> <p>3.2 Verifica en la BD que no existe una respuesta registrada con anterioridad en la región seleccionada para resolver.</p> <p>3.3 Retorna al formulario Resolver ejercicio mostrando un mensaje de éxito de la operación: <i>“Respuestas insertadas en la región”</i>.</p> <p>3.4 Termina el caso de uso.</p> |
| Flujos alternos (Escenario registrar) | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | En el paso 3 del flujo normal de eventos intenta registrar una respuesta en una región que ya la tiene y presiona el botón Enviar. | 1.1 Retorna el formulario Registrar ejercicio mostrando un mensaje de error indicando: <i>“Ya fueron insertadas las respuestas en esa región”</i> . |
| 2. | En el paso 3 del flujo normal de eventos | 2.1 Retorna el formulario Registrar |

| | deja alguno de los campos vacíos y presiona el botón Enviar. | ejercicio mostrando un mensaje de error indicando: <i>“Debes llenar todos los campos”</i> . |
|---|---|--|
| 3. | En el paso 3 del flujo normal de eventos no selecciona una región a resolver. | 3.1 Retorna el formulario Registrar ejercicio mostrando un mensaje de error indicando: <i>“Debes seleccionar una región a resolver”</i> . |
| Flujo de eventos | | |
| Flujo básico “ Escenario eliminar ” | | |
| No | Acción del Actor | Respuesta del Sistema |
| 2. | Selecciona la opción Gestionar Respuestas del menú principal. | 1.1 Muestra el formulario con el listado de los ejercicios registrados y las opciones Resolver y Eliminar Respuesta. |
| 3. | Selecciona la región que desea eliminar sus respuestas y presiona el botón  presente en el lateral derecho del ejercicio que se desea eliminar. (Alterno 1). | 2.1 Elimina todas las respuestas registradas en la base de datos del ejercicio seleccionado. 2.2 Retorna a la interfaz que muestra el listado de ejercicios registrados. 2.3 Termina el caso de uso. |
| Flujos alternos (Escenario eliminar) | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | En el paso 2 del escenario eliminar el actor no selecciona una región para eliminar las respuestas y presiona el botón  . | 1.1 Retorna al formulario Listado de ejercicios registrados mostrando un mensaje de error indicando: <i>“Selecciona una región a eliminar”</i> . |
| Poscondiciones | | Se le muestra al usuario cada una de las interfaces de registrar o eliminar una respuesta de ejercicio. |

Anexo 21:

Tabla 6 Especificación de caso de uso: Asignar ejercicio

| Caso de Uso: | CU_ Asignar ejercicio. | |
|---|--|--|
| Objetivo | Gestionar cada una de las asignaciones de los ejercicios registrado en la aplicación a cada estudiante. | |
| Actores | Profesor | |
| Resumen | El caso de uso inicia cuando el profesor de Física lleva a cabo todo el proceso de asignación de los distintos ejercicios registrados en la aplicación a cada uno de sus estudiantes según el grupo que atiende. | |
| Complejidad | Baja. | |
| Requisitos de Referencia | RF 7. | |
| Prioridad | Crítico. | |
| Precondiciones | El actor debe estar autenticado como Jefe de la asignatura de Física o profesor y acceder a la opción de Asignar Ejercicio. | |
| Flujo de eventos | | |
| Flujo básico “Asignar Ejercicio” | | |
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Selecciona la opción Asignar Ejercicio del menú principal. | 1.1 Muestra el listado de estudiantes que el profesor autenticado atiende. Mostrando el nombre, los apellidos y el grupo. 1.2 Muestra además los botones Asignar y/o Eliminar asignación, presentes ambos en el lateral derecho de cada estudiante. |
| 2. | Presiona el botón Asignar  a un estudiante en específico. | 2.1 Muestra el listado de ejercicios registrados en la aplicación que podrán ser asignados al estudiante. |

| | | |
|-----------------------|---|--|
| 3. | Asigna el ejercicio al estudiante seleccionado presionando el botón Asignar del lateral derecho del ejercicio. | 3.1 Verifica que el ejercicio no le fue asignado al estudiante con anterioridad. 3.2 Retorna a la interfaz que muestra el listado de estudiantes que el profesor autenticado atiende, mostrando un mensaje de éxito de la operación: <i>"Ejercicio asignado correctamente"</i> . 3.3 Termina el caso de uso. |
| Poscondiciones | Se le muestra al usuario cada una de las interfaces de asignación y selección de ejercicios para cada estudiante. | |

Anexo 22:

Tabla 7 Especificación de caso de uso: Generar reporte

| | |
|---------------------------------|--|
| Caso de Uso: | CU_ Generar reporte. |
| Objetivo | Generar un reporte con todas las evaluaciones que obtuvo el estudiante en cada intento por cada paso. |
| Actores | Profesor |
| Resumen | El caso de uso inicia cuando el profesor de Física lleva a cabo el proceso de generar un reporte por cada estudiante de las distintas evaluaciones que fue obteniendo en los intentos de resolución de los ejercicios que le fueron asignados. El mismo será generado en formato .pdf. |
| Complejidad | Media. |
| Requisitos de Referencia | RF 9. |
| Prioridad | Crítico. |
| Precondiciones | El actor debe estar autenticado como Jefe de la asignatura de Física o profesor y acceder a la opción de Generar Reporte. |
| Flujo de eventos | |

| Flujo básico "Generar Reporte" | | |
|--------------------------------|--|---|
| No | Acción del Actor | Respuesta del Sistema |
| 1. | Selecciona la opción Generar Reporte del menú principal. | 1.1 Muestra el listado de estudiantes que el profesor autenticado atiende. Mostrando el nombre, los apellidos y el grupo. 1.2 Muestra además el botón evaluaciones en el lateral derecho de cada estudiante. |
| 2. | Presiona el botón evaluaciones  de un estudiante en específico. | 2.1 Muestra el listado de ejercicios que el estudiante tiene asignado. |
| 3. | Selecciona el ejercicio que desea mostrar sus evaluaciones presionado en botón  . | 3.1 Muestra una interfaz con todas las evaluaciones obtenida por el estudiante en ese ejercicio y por cada uno de los intentos un resultado final. 3.2 Muestra demás la opción de generar el reporte con todas las evaluaciones. |
| 4. | Presiona el botón Generar Reporte. | 4.1 Carga los resultados de las tres posibles regiones a evaluar y los muestra en el pdf que posteriormente será salvado por el profesor. 4.2 Termina el caso de uso. |
| Poscondiciones | | Se le muestra al usuario cada una de las interfaces, que muestran los ejercicios orientados y las evaluaciones obtenidas por el mismo en cada intento de resolución. |

Anexo 13: Modelo de encuesta aplicada a los estudiantes de 2do año de la Facultad 5 para medir el nivel de satisfacción con el uso de la aplicación.

Cuestionario de preguntas:

1. ¿Los ejercicios orientados cumplen con los requisitos para su realización?
2. ¿La forma de evaluación a partir de los pasos definidos es de su agrado?
3. ¿La interfaz de cumple con los requerimientos para la resolución de los ejercicios?
4. ¿Le gustaría que se incluyera el sistema como una herramienta de apoyo al aprendizaje en el EVA?
5. ¿Crees que el uso de la aplicación es importante para el contenido tratado?
6. ¿Una vez utilizada la aplicación, aumentaría su motivación por la asignatura?

Resultados obtenidos:

Nivel de satisfacción:

