

Universidad de las Ciencias Informáticas

Facultad 5



**Propuesta de análisis y diseño de un laboratorio virtual
de Química**

**Trabajo de diploma para optar por el título de Ingeniero
en Ciencias Informáticas**

Autor: *Yaima Laugart Pérez.*
Tutor: *Ing. Sailyn Salas Hechavarria*

La Habana, junio 2012
“Año 53 de la Revolución”

“Las ideas nacen de los conocimientos y de los valores éticos. Una parte importante del problema estaría resuelta tecnológicamente, la otra hay que cultivarla sin descanso o de lo contrario se impondrán los instintos más primarios”.

“La tarea que los graduados de la UCI tienen por delante es grandiosa. Espero que la cumplan, y la cumplirán”.

Fidel Castro Ruz

Declaración de autoría

Declaro que soy el único autor de este trabajo y autorizo a la Facultad 5 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

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

Yaima Laugart Pérez

(Firma del autor)

Ing. Sailyn Salas Hechavarria

(Firma del tutor)

Ing. Roxana Martínez Ferrer

(Firma de la co-tutora)

Agradecimientos

A mi novio Ramiro González Almarales que siempre ha estado presente en los momentos buenos y malos de mi vida.

A Adisleidys Mirabal Pérez por toda su cooperación.

A mi tutora Sailyn Salas por corregirme y guiarme en esta investigación.

Agradezco a mis tías Mabel, Maricela y Maribel por todo su apoyo incondicional cada vez que las necesitaba.

A Orlando por estar siempre presente en los momentos difíciles.

A la Revolución, especialmente a nuestro comandante en Jefe Fidel Castro Ruz por crear nuestra querida Universidad, que se convirtió durante todos estos años en mi hogar y por las incontables experiencias aquí vividas.

A mis queridos amigos de todos estos años que se han convertido en parte de mi familia.

Arianna Bolet, Enelis Cuba, Lauren San Juan, Darlen Martel, Yolanda Fresneda, Nellis Gutiérrez, Ernesto Villalón, Yuri Víctor, Julio Cesar y Alberto Dávila.

¡A todos muchas gracias!

Dedicatoria

Este trabajo lo dedico a mi madre Marlenis y a mi padre Pedro por quererme tanto, por ser mi guía y mi razón de ser, por apoyarme en estos cinco años.

A mis hermanos Yasmani y Mayrelis por quererme como tal.

A mis tías y tíos por creer en mí, por apoyarme en estos cinco años.

Le dedico esta tesis a mi segunda mama, mi tía Melba por apoyarme en todo momento de mi vida, por estar presente siempre que la necesitaba.

A mi abuela y abuelo por apoyarme en este camino tan largo.

A mi abuela Gloria que donde quiera que este sé que me está viendo y guiando por el buen camino, a ti abuela gracias.

A mi novio Ramiro por su apoyo incondicional en todo este tiempo que ha estado a mi lado.

Resumen

La industria del software en el país ha tenido un avance significativo, cada vez se producen más software que apoyan varias esferas de la sociedad. La educación es una de las esferas donde más se han aplicado estos avances, que con el desarrollo de las Tecnologías de la Información y las Comunicaciones posibilitan el desarrollo de habilidades para la construcción del conocimiento de los estudiantes. Los laboratorios virtuales son un ejemplo de herramientas de apoyo al aprendizaje, los cuales son utilizados como una alternativa para la comprensión de algunas asignaturas que se ven afectadas debido a la falta de presupuesto en los laboratorios tradicionales.

Hoy en día en las secundarias básicas del país, no existe un Laboratorio Virtual que cuente con las características necesarias para que el alumno se relacione con el instrumental y los experimentos en las asignaturas que así lo requieran, específicamente para la asignatura Química.

El objetivo de este trabajo es diseñar un bloque de prácticas de laboratorios virtuales, como medio de enseñanza para la asignatura Química en 8vo grado en las secundarias básicas del país, con el fin de propiciar ambientes de aprendizaje alternativos que promuevan el desarrollo de habilidades permitiendo a los estudiantes experimentar sin riesgo alguno y repetir la práctica tantas veces como sea necesario.

En el cuerpo del documento se encontrará un estudio exhaustivo del estado del arte donde se introducen conceptos básicos asociados a los laboratorios virtuales, además se seleccionan las metodologías, herramientas y tecnologías a utilizar. Se definen y especifican las funcionalidades que debe cumplir la herramienta y se modela el sistema. Por último se validó la propuesta de diseño mediante los prototipos de interfaz de usuario.

Tabla de contenido

Introducción	1
Capítulo 1: Fundamentación Teórica	5
Introducción	5
1.1 Los laboratorios virtuales y sus definiciones	5
1.2 Antecedentes históricos de los Laboratorios Virtuales	6
1.3 Ventajas e inconvenientes de los Laboratorios Virtuales	8
1.3.1 Ventajas.....	8
1.3.2 Inconvenientes.....	8
1.4 Razones para contar con prácticas de LV de Química en las secundarias	8
1.5 Metodologías de desarrollo.....	9
1.5.1 XP.....	9
1.5.2 RUP.....	10
1.5.3 SCRUM	11
1.5.4 Selección de la metodología a emplear en la solución propuesta	12
1.6 Lenguaje de modelado	12
1.7 Herramientas CASE	13
1.7.1 Rational Rose Enterprise Edition 2003	13
1.7.2 Visual Paradigm.....	14
1.7.3 Selección de la herramienta para el análisis y diseño del sistema	14
1.7.4 Entorno de Desarrollo Integrado	15
1.7.5 Eclipse.....	15
1.7.6 QT Creator.....	15
1.7.7 Selección del Entorno de Desarrollo Integrado	15
1.8 Motor Gráfico. Object Oriented Graphics Engine (Ogre).....	16
1.9 Herramientas de diseño gráfico.	16
1.9.1 Autodesk 3ds Max y Autodesk 3ds Max Design.....	16
1.9.2 Blender.....	17
1.9.3 Selección de las herramientas de diseño gráfico	17
1.10 Sistemas Gestores de Bases de Datos.....	17
1.10.1 MySQL.....	18
1.10.2 PostgreSQL	18
1.10.3 SQLite.....	18
1.10.4 Selección del SGBD a utilizar	19
Consideraciones Parciales.....	19
Capítulo 2: Características del sistema.	20
2.1 Modelo del negocio.....	20
2.2 Descripción del negocio actual	20
2.3 Reglas del negocio	21
2.4 Modelo de Casos de Uso del negocio	21
2.4.1 Actores del negocio	21

2.4.2	Caso de uso del negocio	22
2.4.3	Diagrama de casos de uso del negocio	22
2.4.4	Modelo de objetos del negocio	23
2.4.5	Trabajador del negocio.	23
2.4.6	Especificación de los casos de uso del negocio.....	23
2.4.7	Diagrama de Actividades	24
2.5	Requisitos.....	25
2.5.1	Requisitos Funcionales.....	25
2.5.2	Requisitos no Funcionales.....	27
2.6	Modelos de casos de uso del sistema	28
2.6.1	Actores del Sistema.....	28
2.6.2	Casos de uso del sistema.....	29
2.6.3	Diagrama de Casos de Uso del Sistema.....	29
2.6.4	Descripción de casos de uso.	30
	Consideraciones Parciales.....	41
	Capítulo 3: Análisis y Diseño del Sistema	42
	Introducción	42
3.1	Patrones de Arquitectura	42
3.2	Patrón Modelo-Vista-Controlador (MVC):	42
3.2.1	Patrones de diseño aplicados	43
3.3	Análisis y Diseño	44
3.3.1	Diagrama de Clases del Análisis	44
3.3.2	Diagrama de Interacción.....	46
3.3.2.1.	Diagramas de Colaboración.....	46
3.3.3	Diagrama de Clases del Diseño.....	49
3.3.3.1.	Diagrama de Secuencia del Diseño	50
3.3.4	Diagrama de Despliegue	53
3.3.5	Diagrama de clases persistentes	54
3.3.6	Modelo de Datos.....	54
3.4	Validación de la propuesta de diseño.	55
3.4.1	Propósitos del Prototipo.....	55
3.4.2	Clases de Prototipo	55
3.4.3	Prototipo no funcional	55
	Consideraciones Parciales.....	60
	Conclusiones generales.....	62
	Recomendaciones	63
	Bibliografía.....	64
	Anexos.....	68
	Diagramas de Clases del Análisis	68
	Diagramas de Colaboración.....	70
	Diagramas de Secuencias del Diseño.....	75

Introducción

El avance de las Tecnologías de la Información y las Comunicaciones (TICs) ha provocado cambios en diversas áreas del conocimiento. El sector de la educación es privilegiado en este sentido, ya que cada día se crean nuevas herramientas educativas con el objetivo de apoyar el proceso de enseñanza aprendizaje. Los cambios en esta esfera están relacionados con la manera en que el estudiante recibe y demuestra los conocimientos adquiridos, además de la forma en que los profesores imparten clases. Una de las formas de docencia que posibilitan el afianzamiento de los conocimientos teóricos y el desarrollo de habilidades manuales son los laboratorios virtuales. La asignatura de Química, por sus características, necesita del uso de este tipo de laboratorio, dada la diversidad de experimentos y demostraciones que posee.

Se puede afirmar que los ambientes de aprendizaje mediados por las TICs desempeñan un papel muy importante en la enseñanza de la química, posibilitando a profesores y alumnos realizar experimentos y desarrollar proyectos en laboratorios con la utilización de computadoras y dispositivos diversos, utilizando el programa de estudio, provisto por el profesor.

Una situación en la que los profesores se relacionan con las tecnologías es al utilizar entornos virtuales en el proceso de formación docente, ejemplo de esto son los laboratorios virtuales los que se utilizan como herramientas de apoyo en el estudio de situaciones reales. Además de ser una excelente herramienta desde el punto de vista pedagógico y en muchos aspectos, un ámbito esencial para la enseñanza de las ciencias, dando la posibilidad de aprender a partir de sus propias experiencias, estimulando la curiosidad y el placer por la investigación. En el mundo se ha incrementado el número de proyectos de este tipo, la mayoría en la esfera educativa.

Las Prácticas de Laboratorios Virtuales (PLV) le permiten al alumno vivir la experiencia de utilizar sustancias tóxicas, de uso común en el laboratorio de química experimental, dentro de un ambiente protegido, donde la seguridad y la preservación de la salud humana son fundamentales, ya que se presentan escenarios de aprendizaje donde es necesario que el alumno resuelva situaciones semejantes a las de una experiencia real, preparándolo para desenvolverse exitosamente en un laboratorio real.

El Laboratorio Virtual (LV) es considerado una alternativa para desarrollar habilidades. Sin embargo, debemos ser conscientes de que el experimento virtual no incide de la misma

manera que lo vivencial o lo real, por tanto puede ser una herramienta para aquellos centros docentes que no cuentan con las condiciones adecuadas para realizar la actividad experimental. Los laboratorios virtuales no sustituyen al laboratorio real o tradicional, sólo sirven de apoyo al proceso docente educativo.

Situación problemática:

- En las secundarias básicas de nuestro país, específicamente en 8vo grado, no se cuenta con los recursos necesarios para que los estudiantes puedan llevar a cabo las prácticas de los laboratorios de Química.
- Las limitaciones materiales en relación con los laboratorios reales tienen como resultado final, la mayoría de las veces, a un profesor realizando demostraciones y no a los alumnos experimentando por sí mismos; y si la actividad no se desarrolla satisfactoriamente, rara vez puede repetir el experimento debido al escaso material.
- Se han producido un gran número de laboratorios dirigidos a la docencia en el mundo, que simulan los procesos naturales, en este caso Química, pero su precio en el mercado es elevado y de difícil acceso.
- Cabe destacar que estas limitaciones dificultan, la comprensión de conceptos imposibles de observar a simple vista, el desarrollo de habilidades de aprender por cuenta propia y la construcción del conocimiento partiendo de la capacidad de análisis, síntesis y evaluación del propio estudiante, que sólo se podría resolver a través de la solución de problemas propuestos en los laboratorios digitales y el uso de las tecnologías.

Se formula como **Problema Científico:**

¿Cómo contribuir al desarrollo de conocimientos y habilidades experimentales de los estudiantes de secundaria en la asignatura de Química?

Como **objeto de estudio:** Los Laboratorios Virtuales.

Para dar solución al problema planteado se define como **objetivo general:** Diseñar un bloque de prácticas de laboratorios virtuales, como medio de enseñanza para la asignatura Química en 8vo grado en las secundarias básicas del país.

Todo lo anterior precisa como **campo de acción**: Los Laboratorios Virtuales de Química.

Se plantea como **idea a defender**: Con la propuesta de análisis y diseño de un Laboratorio Virtual de Química se contribuye a la elaboración de un medio de enseñanza que permita la realización virtual de las prácticas de laboratorios de esa asignatura, con ahorro de recursos.

Para satisfacer el objetivo general planteado anteriormente se proponen las siguientes **tareas de investigación**:

1. Revisión de la documentación relacionada con el tema a investigar en el mundo, Cuba y la UCI.
2. Identificación de los aspectos a favor y en contra de un Laboratorio Virtual de química.
3. Análisis de las prácticas de laboratorios a desarrollar.
4. Análisis de las metodologías de desarrollo de software, herramientas de modelado, lenguajes de programación y entornos de desarrollo para seleccionar los más adecuados a utilizar.
5. Definición de los requisitos funcionales y no funcionales para plasmar las funcionalidades y restricciones que tiene el sistema a desarrollar.
6. Elaboración del modelo de diseño para definir la vista interna del sistema.
7. Validación del diseño que se propone a través de un demo.

Para dar solución a las tareas antes mencionadas se utilizaron los siguientes **métodos de investigación**:

Métodos teóricos:

- **Analítico–Sintético**: se utilizó en el estudio de la base teórica y de las herramientas y tecnologías a emplear, para definir las características esenciales de las mismas y sus relaciones.
- **Análisis Documental**: facilitó el análisis y la sistematización del tema sobre LV a partir de documentos de autores extranjeros y nacionales.
- **Histórico–Lógico**: para estudiar las tendencias históricas y actuales en el desarrollo de Laboratorios Virtuales en el mundo, Cuba y la UCI; así como determinar su incidencia en el problema actual de la investigación.
- **Modelación**: para la mejor comprensión del problema planteado en la investigación utilizando modelos que tienen una estrecha relación con el objeto que se estudia, estos modelos representan las posibles soluciones que tendrá el problema.

Métodos empíricos:

- **Entrevista:** permitió la exploración y constatación del estado actual en la concepción, producción y utilización de LV.

Estructura del documento:

Capítulo 1: Fundamentación Teórica. Se fundamentan los elementos teóricos que sustentan la investigación y se realiza un estudio sobre laboratorios virtuales existentes. Se hace referencia a las diferentes herramientas y tecnologías utilizadas en el desarrollo del LV, las cuales dan solución al problema planteado.

Capítulo 2: Características del sistema. Contiene un marco conceptual asociado a la información que será manipulada por el sistema, descripción de los procesos del negocio, modelo de negocio, especificación de los requisitos de software y definición de los casos de uso.

Capítulo 3: Análisis y diseño del sistema. Se presentan los diagramas de clases del análisis y el diseño, que reflejan de una forma más clara cómo va a funcionar el sistema y qué clases estarán presentes en el mismo; así como la validación de los resultados a través de un prototipo no funcional.

Capítulo 1: Fundamentación Teórica

Introducción

En el presente capítulo se analizan algunas definiciones sobre los Laboratorios Virtuales. Se elabora el marco teórico de la investigación, se caracteriza la evolución de los mismos al conocer los antecedentes históricos que los originaron, analizando sus ventajas y desventajas. Así como la necesidad de contar con un LV de Química en las secundarias básicas del país. Se realiza un estudio sobre las principales metodologías de desarrollo de software, las herramientas y tecnologías que se utilizan en la actualidad; en dependencia de las características de cada una se selecciona la más adecuada para la propuesta.

1.1 Los laboratorios virtuales y sus definiciones

“Los laboratorios virtuales se enmarcan en lo que se conoce como entornos virtuales de aprendizaje (EVA) que, aprovechando las funcionalidades de las TICs, ofrecen nuevos entornos para la enseñanza y el aprendizaje libres de las restricciones que imponen el tiempo y el espacio en la enseñanza presencial y capaces de asegurar una continua comunicación (virtual) entre estudiantes y profesores” (1).

Según bibliografía consultada, son muchos los autores que han dado una definición sobre las principales ideas referentes al LV, a continuación se citan algunos ejemplos:

“Para trabajar sobre los procesos de la ciencia, habría que destacar, dentro del software específico, los laboratorios virtuales, que permiten desarrollar objetivos educativos propios del trabajo experimental. Se entiende por laboratorio virtual un sitio informático que simula una situación de aprendizaje propia del laboratorio tradicional.” (1)

“Un Laboratorio Virtual es un conjunto de recursos compartidos en la red (un cuaderno de notas digital, ficheros, búsquedas) con el fin de que los usuarios puedan poner en práctica, mediante el control remoto, la monitorización de los experimentos y la gestión de dichos recursos, los conocimientos adquiridos en las aulas de las Universidades sin tener que contar con material sofisticado o con componentes caros y difíciles de obtener.” (2)

Características de los laboratorios virtuales

“Estos laboratorios, aplicados a la enseñanza secundaria, permiten:

- Simular un laboratorio de ciencias que permita solucionar el problema de equipamiento, materiales e infraestructura de los laboratorios presenciales.
- Recrear procesos y fenómenos imposibles de reproducir en un laboratorio presencial e intervenir en ellos.
- Desarrollar la autonomía en el aprendizaje de los estudiantes.
- Tener en cuenta las diferencias en el ritmo de aprendizaje de los alumnos a un nivel más profundo de lo que es posible en el laboratorio presencial (posibilidad de repetir las prácticas o alterar su secuencia, por ejemplo).
- Desarrollar en los estudiantes habilidades y destrezas en el uso de las TIC.
- Desarrollar una nueva forma de aprendizaje que estimule en los estudiantes el deseo por aprender e investigar.
- Incluir sistemas de evaluación que permitan ajustar las ayudas pedagógicas a las necesidades de los alumnos.
- Sustituir al profesor en las tareas más rutinarias, como la exposición de conceptos, permitiéndole dedicar más tiempo a los alumnos individualmente.” (1)

El LV, se puede ubicar como una nueva alternativa a los simuladores y juegos educativos, los cuales cuentan con un escenario en tercera dimensión con una serie de componentes que conforman el laboratorio, un dispositivo de interacción (guante, casco, traje, y otros), que permite al estudiante introducirse dentro de un mundo virtual y un dispositivo de captura de movimiento que recibe las señales de movimiento del estudiante dentro del mundo virtual y las lleva a computador para ser procesadas. (3)

Basándose en el análisis de los temas estudiados y las definiciones dadas en párrafos anteriores, la autora define como Laboratorio Virtual:

Son programas informáticos que permiten simular experimentos propios de laboratorios reales, permitiendo a los estudiantes reafirmar sus conocimientos y evaluarse sin exigir la presencia de un profesor.

1.2 Antecedentes históricos de los Laboratorios Virtuales

La primera aproximación a los laboratorios virtuales aparece en el año 1984, donde surge el concepto de instrumento virtual y sus características se determinaron de acuerdo a los fundamentos de programación. Durante los años posteriores se fueron lanzando distintas propuestas para laboratorios, entre ellas la de un laboratorio de control de sistemas en 1991 en

la Universidad de Bucknell en E.E.U.U., que cuenta con un sistema de procesado digital de señal y conexión a Internet.

En 1992 aparece el término LV, para describir la programación orientada a objetos en el desarrollo de un laboratorio de simulación. En 1994 se presenta un estudio realizado por la Universidad de Vanderbilt en E.E.U.U. en el que se desarrolla un laboratorio virtual basado en simulación como apoyo a las prácticas tradicionales y, que concluyó, con la necesidad de esta herramienta para aprender las habilidades básicas y el manejo de los equipos, lo cual optimizaba tanto el tiempo de los alumnos como el del personal de laboratorio. En 1994 aparece un artículo en el que se define explícitamente un LV como un programa de simulación. Ya en la conferencia IMTC (IEEE Instrumentation and Measurement Technology Conference) celebrada en junio de 1996, empiezan a presentarse distintos aspectos de lo que es un LV. Desde entonces las referencias de los laboratorios se han incrementado vertiginosamente.” (3)

Investigadores de la Universidad de Illinois en 1997 presentan un completo laboratorio de instrumentación electrónica puesto a disposición de los usuarios a través de Internet. Este es el primer LV con control remoto de instrumentación electrónica en funcionamiento.

Primeros pasos en Cuba

La Universidad Central “Marta Abreu” de Villa Clara en Cuba presentó en el 2004 un material didáctico interactivo que permite la realización de las prácticas de laboratorio de Física General, a los alumnos de las carreras técnicas de los centros de educación superior, desde un computador.

La Universidad de las Ciencias Informáticas (UCI) es otro de los centros que en la actualidad se encarga a nivel nacional de la producción de Software Educativo, incluyendo dentro de estos los Entornos Virtuales de Aprendizaje(EVA),no son pocos los trabajos de investigación realizados con el objetivo de crear laboratorios virtuales para diferentes niveles de enseñanza y teniendo en cuenta la demanda de determinadas instituciones, tal es el caso del Proyecto Laboratorios Virtuales (PROLAVI) el cual desarrolló un LV de ensamblaje de un computador para apoyar la unidad curricular Arquitectura de un Computador y otro de Configuración y Administración de una Red LAN I y II para apoyar la unidad curricular “Redes de un computador” en las universidades de la República Bolivariana de Venezuela en el 2011.

1.3 Ventajas e inconvenientes de los Laboratorios Virtuales

1.3.1 Ventajas

Estos laboratorios, aplicados a la enseñanza secundaria, permiten:

- “Simular un laboratorio que permita solucionar el problema de equipamiento, materiales e infraestructura de los laboratorios presenciales.
- Recrear procesos y fenómenos imposibles de reproducir en un laboratorio presencial e intervenir en ellos.
- Desarrollar en los estudiantes habilidades y destrezas en el uso de las TIC.
- Desarrollar una nueva forma de aprendizaje que estimule en los estudiantes el deseo por aprender e investigar.
- Incluir sistemas de evaluación que permitan ajustar las ayudas pedagógicas a las necesidades de los alumnos.
- Sustituir al profesor en las tareas más rutinarias, como la exposición de conceptos, permitiéndole dedicar más tiempo a los alumnos individualmente”. (1)

1.3.2 Inconvenientes

Además de ventajas también existen inconvenientes en este proceso tales como:

- “No puede sustituir la experiencia práctica altamente enriquecedora del Laboratorio Tradicional (LT).
- Se corre el riesgo que el estudiante se comporte como un mero espectador. Es importante que las actividades en el LV, vengan acompañadas de un guión que explique el concepto a estudiar, así como las ecuaciones del modelo utilizado. Es necesario que el estudiante realice una actividad ordenada y progresiva, conducente a alcanzar objetivos básicos concretos.
- En el LV el estudiante no utiliza elementos reales, lo que provoca una pérdida parcial de la visión de la realidad. Además, no siempre se dispone de la simulación adecuada para el tema que el profesor desea trabajar”. (4)

1.4 Razones para contar con prácticas de LV de Química en las secundarias

Las condiciones económicas actuales del país y a su vez del mundo, dificultan el desarrollo de prácticas de la asignatura de Química en un laboratorio tradicional por lo que, los siguientes argumentos justifican la necesidad de crear y utilizar Prácticas de Laboratorios Virtuales de

Química (PLVQ) atendiendo las características de los estudiantes de las secundarias básicas deberían ser:

- La disciplina Química requiere de prácticas de laboratorio para consolidar los conocimientos adquiridos en las clases.
- Simular un laboratorio que permita solucionar el problema de equipamiento, materiales e infraestructura de los laboratorios presenciales.
- Las secundarias no cuenta con PLVQ propias, que permitan al estudiante entrar en un entorno virtual donde tengan la libertad de efectuar elecciones y tomar decisiones como las que afrontarían en un medio real de laboratorio.
- No cuenta la secundaria con prácticas virtuales de Química que respondan al programa de la asignatura que se imparte en las secundarias de nuestro país.
- Al interactuar los estudiantes con la simulación comprenden mejor los sistemas, procesos o fenómenos reales explorando conceptos, comprobando hipótesis o descubriendo explicaciones.

1.5 Metodologías de desarrollo

Una metodología es un conjunto de procedimientos, técnicas, herramientas y un soporte documental que ayuda a realizar un software. Para la misma se definen tareas que constituyen las actividades elementales en que se dividen los procesos y permiten ser ejecutadas a través de los procedimientos. En este sentido se definen técnicas que se pueden utilizar una o varias veces y herramientas software que automatizan la aplicación. (5)

En estos últimos años se han desarrollado dos corrientes en lo que respecta a los procesos de desarrollo. Por un lado se encuentran las metodologías tradicionales o pesadas entre las que se encuentran el Rational Unified Process (RUP), mientras que por el otro están las metodologías ágiles de las cuales se pueden mencionar Extreme Programming (XP), SCRUM. La diferencia fundamental entre ellas está reflejada en que mientras las tradicionales intentan lograr el objetivo común por medio de orden y documentación, las ágiles tratan de mejorar la calidad del software a través de una comunicación directa e inmediata entre las personas que intervienen en el proceso. (5)

1.5.1 XP

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose

por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo. Se basa en retroalimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. Es definida especialmente como una metodología adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. (6)

Esta metodología intenta reducir la complejidad del SW por medio de un trabajo orientado directamente al objetivo, basado en las relaciones interpersonales y la velocidad de reacción. Presenta una debilidad cuando se utiliza en dominios de aplicaciones complejas o situaciones difíciles en la organización: el rol del cliente no refleja los diferentes intereses, habilidades y fuerzas a las que se enfrentan los programadores durante el desarrollo de proyectos. Además, presenta un diseño evolutivo que hace que se le de poca importancia al análisis como fase independiente, puesto que se trabaja exclusivamente en función de las necesidades del momento. (7)

1.5.2 RUP

Es una metodología de desarrollo de software que está basada en componentes e interfaces bien definidas. Es un proceso que puede especializarse para una gran variedad de sistemas de software, en diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto.

Como características presenta la utilización del UML como lenguaje de modelado, unifica los mejores elementos de metodologías anteriores, se encuentra preparado para desarrollar grandes y complejos proyectos y es orientado a objetos.

RUP propone 9 flujos de trabajo que se encuentran insertados en 2 disciplinas: Disciplina de Desarrollo (Ingeniería de Negocios, Requerimientos, Análisis y Diseño, Implementación y Pruebas) y Disciplina de Soporte (Configuración y administración del cambio, Administración del proyecto, Ambiente y distribución). Su ciclo de vida se caracteriza por ser:

- Dirigido por Casos de Uso: Los casos de uso describen los requisitos funcionales del sistema desde la perspectiva del usuario. Estos no solo inician el proceso de desarrollo sino que también proporcionan un hilo conductor, en el que se verifica, luego de la implementación, que el producto implemente adecuadamente cada Caso de Uso.
- Centrado en la arquitectura: La arquitectura de un sistema permite tener una visión común entre los desarrolladores y los usuarios. Por otra parte, la arquitectura es una

vista del diseño completo con las características más importantes resaltadas, dejando los detalles de lado.

- Iterativo e incremental: RUP propone un proceso iterativo e incremental donde el trabajo se divide en partes más pequeñas. Cada una de estas partes se puede ver como una iteración de la cual se obtiene un incremento que produce un crecimiento en el producto. (8)

1.5.3 SCRUM

SCRUM es una metodología ágil de desarrollo que surge como modelo para el desarrollo de productos tecnológicos, aunque también se emplea en entornos que trabajan con requisitos inestables y que requieren rapidez y flexibilidad, situaciones frecuentes en el desarrollo de determinados sistemas de software. Esta metodología requiere trabajo duro puesto que no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto. Además, en ocasiones es necesario complementarlo con otras metodologías.

Como metodología ágil:

- Es un modo de desarrollo de carácter adaptable más que predictivo.
- Orientado a las personas más que a los procesos.
- Emplea la estructura de desarrollo ágil: incremental basada en iteraciones y revisiones.

Además, emplea las siguientes prácticas de la gestión ágil:

- Revisión de las iteraciones
- Desarrollo incremental
- Desarrollo evolutivo
- Auto-organización
- Colaboración

Sus principales características se pueden resumir en dos. El desarrollo de software se realiza mediante iteraciones, denominadas sprints, con una duración de 30 días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente. La segunda característica importante son las reuniones a lo largo del proyecto, entre ellas destaca la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración. (9)

1.5.4 Selección de la metodología a emplear en la solución propuesta

En la actualidad no existe, según la bibliografía consultada una metodología de desarrollo de software que sea global, es decir que encierre características que puedan aplicarse a cualquier tipo de proyecto. Las características de cada proyecto conjuntamente con su equipo de desarrollo, recursos, y requisitos exigen que se escoja una que se adapte en la mayor medida posible.

Teniendo en cuenta las necesidades del laboratorio a desarrollar, se llega a la conclusión que la metodología más adecuada para dirigir el proceso de desarrollo del software es RUP, a pesar de estar recomendada para proyectos extensos y grandes grupos de trabajo, esta define claramente actividades realizadas por roles generando a su paso artefactos que sustentan el proceso de desarrollo del producto.

RUP además constituye una metodología robusta y como tal basa su éxito en mantener una documentación ordenada a lo largo de todo el proceso de desarrollo, siendo este último un aspecto importante, teniendo en cuenta que el personal del proyecto es muy cambiante y muchas veces de poca experiencia, por lo que la documentación permitirá que no se pierdan elementos del producto que pueden ser de utilidad para su continuidad. Además el lenguaje utilizado por RUP para la modelación del sistema es UML, el cual brinda amplias posibilidades en la representación, es de fácil uso y conocido por el equipo de proyecto.

1.6 Lenguaje de modelado

Lenguaje Unificado de Modelado (UML) es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables. UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas. (10)

“El Lenguaje Unificado de Modelación, proporciona ventajas en la representación del ciclo de vida de un software y de los artefactos específicos del Proceso Unificado de Desarrollo del Software. Permite una comunicación sencilla y rápida entre desarrolladores y clientes del software que se desarrolla.” (11)

1.7 Herramientas CASE

La herramienta CASE es aquella que permite crear los artefactos necesarios siguiendo una metodología en la construcción de un software. Las herramientas constituyen un componente básico en el desarrollo de cualquier proyecto ya que pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como:

- “Mejorar la productividad en el desarrollo y mantenimiento del software.
- Aumentar la calidad del software.
- Reducir el tiempo y coste de desarrollo y mantenimiento de los sistemas informáticos.
- Mejorar la planificación de un proyecto
- Aumentar la biblioteca de conocimiento informático de una empresa ayudando a la búsqueda de soluciones para los requisitos.
- Automatizar el desarrollo del software, la documentación, la generación de código, las pruebas de errores y la gestión del proyecto.
- Ayuda a la reutilización del software, portabilidad y estandarización de la documentación
- Gestión global en todas las fases de desarrollo de software con una misma herramienta.
- Facilitar el uso de las distintas metodologías propias de la ingeniería del software.” (12)

1.7.1 Rational Rose Enterprise Edition 2003

Rational Rose es una herramienta de desarrollo basada en modelos que se integra con las bases de datos y los IDE de las principales plataformas del sector. Todos los productos de Rational Rose dan soporte a Unified Modeling Language (UML), pero no son compatibles con las mismas tecnologías de implementación. Rational Rose Enterprise es un entorno de modelado que permite generar código a partir de modelos Ada, ANSI C++, C++, CORBA, Java/J2EE, Visual C++ y Visual Basic. Al igual que todos los productos de Rational Rose, ofrece un lenguaje de modelado común que agiliza la creación del software. Facilita el desarrollo de un proceso cooperativo en el que todos los agentes tienen sus propias vistas de información (vista de Casos de Uso, vista Lógica, vista de Componentes y vista de Despliegue).

Además tiene como características el 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, permite la generación de informes para optimizar la comunicación dentro del equipo. “Es una herramienta que provee sus funciones de una manera más organizada que Visual Paradigm, componentes y estereotipos, son

visualmente más estéticos que los que proporciona Visual Paradigm, es una herramienta muy recomendada y además profesional, pero tiene dos desventajas en su contra, pues obliga al usuario a desarrollar en máquinas con el sistema operativo Windows y a veces resulta difícil el trabajo con ella, ya que hay que efectuar varios pasos para la creación de los artefactos y la incorporación de los componentes”. (13)

1.7.2 Visual Paradigm

Visual Paradigm es una herramienta CASE. La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Ha sido una herramienta UML profesional que soporta el ciclo de vida completo del proceso de desarrollo del software: análisis y diseño orientados a objeto, construcción, pruebas y despliegue. El software de modelado UML ayuda a una rápida construcción de aplicaciones de calidad y a un menor costo. Permite dibujar todos los tipos de diagramas de clases, código inverso, generar código, desde diagramas y generar documentación. Fue diseñado para una amplia gama de usuarios interesados en la construcción de sistemas de software de forma fiable, a través de la utilización de un enfoque Orientado a Objetos. Esta herramienta permite aumentar la calidad del software, a través de la mejora de la productividad en el desarrollo y mantenimiento del software. Entre otros aspectos a destacar se tienen que Visual Paradigm permite la reutilización del software, portabilidad y estandarización de la documentación, además del uso de las distintas metodologías propias de la Ingeniería del Software. Se encuentra disponible en múltiples plataformas (Windows, Linux).El mismo posibilita la generación de bases de datos transformación de diagramas de Entidad-Relación en tablas de base de datos. Admite la exportación e importación de ficheros XML y facilita generar informes. (14)

1.7.3 Selección de la herramienta para el análisis y diseño del sistema

Partiendo del estudio anteriormente realizado se resume que Visual Paradigm ofrece características más relevantes para desarrollar el análisis y diseño del LV, ya que se ajusta a las características que persigue la Universidad de Ciencias Informáticas: es multiplataforma (Windows y Linux). Aporta a los desarrolladores de software una plataforma para construir aplicaciones más baratas y de mejor calidad. Además la universidad cuenta con la licencia de esta herramienta para su utilización. Es importante destacar también que ofrece versiones libres como es el caso de Visual Paradigm UML 6.4 Community Edition.

1.7.4 Entorno de Desarrollo Integrado

Se decide realizar un estudio de los IDEs que más se utilizan en la Universidad de Ciencias Informáticas, específicamente en la Facultad 5, para el desarrollo de sistemas orientados a objetos. Analizar las ventajas e inconvenientes que ofrece cada uno y determinar cuál de estos es más conveniente a emplear, para desarrollar el LV de Química.

1.7.5 Eclipse

Eclipse es un IDE gratuito y de código abierto, multiplataforma (GNU/Linux, Solaris, Mac OSX, Windows) da soporte a varios lenguajes de programación, como son C/C++, PHP o Python (15). Permite añadir extensiones (plugins) a su plataforma base para ampliar sus funcionalidades (16).

Proporciona asistentes (wizards) para la creación, exportación e importación de proyectos; con el objetivo de generar esqueletos de códigos (templates). La compilación es en tiempo real, provee varias herramientas y utilidades que hacen más intuitiva, visual y fácil la programación.

1.7.6 QT Creator

Qt Creator es un excelente IDE basado en la librería Qt, multiplataforma (Linux, Mac OSX, Windows, Windows CE, Symbian y Maemo). Permite desarrollar aplicaciones en C++ de manera sencilla y rápida debido a que es un editor avanzado para dicho lenguaje. Brinda soporte a varios lenguajes, como son: C#/.NET Languages (Mono); Python: PyQt y PySide, Ada, Pascal, Perl, PHP y Ruby. Posibilita el diseño de formularios (GUI) integrado, el completamiento automático y la depuración visual. Cuenta con herramientas para la administración y construcción de proyectos (17).

Facilita aumentar la productividad de los desarrolladores con experiencia en Qt, debido a que se ajusta a sus necesidades. Es un entorno integrado para la creación y diseño de forms para proyectos C++ que permite diseñar rápidamente widgets. Es un entorno de desarrollo integrado de software libre y código abierto. Además el depurador visual (visual debugger) para C ++ es consciente de la estructura de muchas clases de Qt, lo que aumenta la capacidad de mostrar los datos de Qt con claridad (18).

1.7.7 Selección del Entorno de Desarrollo Integrado

Luego de analizar las ventajas que ofrecen ambos IDEs y teniendo en cuenta el desarrollo de otros Laboratorios Virtuales llevados a cabo en la Facultad 5 los cuales han sido

implementados con Qt Creator entorno de desarrollo integrado de software libre y código abierto. Se propone utilizar en la investigación el mismo, porque los desarrolladores están más familiarizados con este entorno. Lo que permitirá la implementación del LV de Química en el menor tiempo posible y con la calidad requerida. Además porque Qt Creator permite diseñar rápidamente widgets y diálogos que se usarán en la aplicación, ofreciendo más completamiento de código a la hora de implementar las funcionalidades del sistema.

1.8 Motor Gráfico. Object Oriented Graphics Engine (Ogre)

El motor gráfico es de código abierto, proporciona funciones de renderizado 2D, 3D y de sprites⁹. Se suele apoyar en bibliotecas/APIs gráficas como OpenGL o DirectX. Es flexible y orientado a objetos, así como la portabilidad que posee en múltiples entornos como Windows, MacOS X o Linux. Ha sido diseñado con la idea de hacer más sencillo el desarrollo de juegos 3D, además de que explota al máximo las posibilidades de las tarjetas gráficas. Partiendo del análisis anterior se selecciona Ogre3D ya posee un manejador de escena altamente flexible y personalizable. Permite así mismo la vinculación con el framework Qt, que es el seleccionado para el diseño de la interfaz gráfica del laboratorio virtual. Además, el equipo de desarrollo de los laboratorios virtuales en la facultad 5 ha realizado exitosos laboratorios usando ese motor.

1.9 Herramientas de diseño gráfico.

Las herramientas de diseño gráfico permiten realizar la modelación y manipulación de objetos y entornos virtuales, brindan un conjunto de funciones que facilitan el trabajo con escenarios virtuales y algunas posibilitan la creación de paseos virtuales.

1.9.1 Autodesk 3ds Max y Autodesk 3ds Max Design.

Proporcionan potentes herramientas integradas de modelado, animación, renderización y composición en 3D que multiplican rápidamente la productividad de los artistas y diseñadores. Aunque ambas versiones comparten la tecnología y la funcionalidad básicas, una ofrece herramientas y experiencias específicas a los desarrolladores de juegos, realizadores de efectos visuales y diseñadores gráficos, mientras que la otra tiene características especializadas para los arquitectos, diseñadores, ingenieros y especialistas en visualización.

(19)

El premiado software Autodesk 3ds Max es la herramienta preferida por los profesionales líderes en desarrollo de juegos, televisión, cine y composición digital que quieren una solución

3D completa con resultados inmediatos. Además proporciona múltiples opciones de renderización integradas, incluida renderización en red ilimitada con tecnología mental ray. (20)

1.9.2 Blender

Es sabido que Blender es una poderosa herramienta para modelado 3D digital, pero que tiene un modo de funcionamiento y una interfaz que son distintas a las de otros programas 3D, que los arquitectos suelen utilizar. De hecho, reúne también amplias funciones para animación, composición, video y juegos. Blender 3D tiene otras características que son ventajosas. La principal es que se trata de Software Libre y además gratuito, condicionando esto, que se pueda descargar e instalar sin ninguna restricción y costo. Otra ventaja es que está disponible en todas las plataformas, permitiendo instalarlo en Windows, Mac OSX o GNU-Linux. (21)

Blender es un programa de modelado y animación con las siguientes capacidades: animación 3D, creación de videojuegos 3D en tiempo real, generación de ambientes y paisajes 3D, generación de efectos especiales 3D, sombreado 3D, texturizado interactivo 3D, modelado 3D y simulación física.

Sin embargo; además de las animaciones y los videojuegos Blender también resulta una herramienta útil para las siguientes actividades: modelado 3D para aplicaciones de arquitectura e industriales, diseño de páginas internet, modelado 3D interactivo de productos para aplicaciones de comercio (E-commerce) y mercadeo (E-marketing) electrónicos y presentaciones 3D interactivas con PowerPoint, Director, Word

1.9.3 Selección de las herramientas de diseño gráfico

Partiendo del estudio anteriormente realizado se resume que Blender ofrece características más relevantes para desarrollar el diseño gráfico, ya que se ajusta a las características que persigue la Universidad de Ciencias Informáticas: es una poderosa herramienta para modelado 3D digital, multiplataforma, de Software Libre y además gratuito

1.10 Sistemas Gestores de Bases de Datos

Los Sistemas Gestores de Bases de Datos (SGBD) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones.

El software libre se ha consolidado como alternativa, técnicamente viable y económicamente sostenible al software comercial, convirtiéndose el software libre como otra alternativa para

ofrecer los mismos servicios a un costo significativamente reducido. Es por ello que se decidió analizar solamente algunos SGBD que son software libre, los más importantes y reconocidos.

1.10.1 MySQL

Sistema Gestor de Bases de Datos (SGBD) relacional, de código abierto y licencia GPL para aplicaciones no comerciales. Es una base de datos segura, fácil de usar y gracias a la colaboración de muchos usuarios, se ha ido mejorando y optimizándose en velocidad (22).

Existe una gran cantidad de software que lo utiliza, es multiusuario, multihilos y un servidor SQL (Structured Query Language). (23)Presenta gran portabilidad entre sistemas; ya que puede trabajar en distintas plataformas y sistemas operativos. Cuenta con un flexible sistema de contraseñas (passwords) y gestión de usuarios, con un muy buen nivel de seguridad en los datos. Es un sistema con baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está. Pero no obstante un gran porcentaje de las utilidades de MySQL no están documentadas

1.10.2 PostgreSQL

El SGBD PostgreSQL proporciona un gran número de características que normalmente sólo se encontraban en las bases de datos comerciales de alto calibre tales como DB2 u Oracle (24). Debido a que es un potente servidor de base de datos relacional, orientado a objetos, de uso libre, multiplataforma que incluye extensa documentación y un enorme manual. Mantiene la integridad de los datos, fiabilidad al respecto de la manutención de la integridad de la información, soportando claves primarias (simples y compuestas). Tiene la capacidad de controlar el acceso según usuario/grupo de usuarios, base de datos a la que se conecta y la autenticación así como la transferencia de datos que es encriptado con SSL. Es capaz de ajustarse al número de Unidades Centrales de Procesamiento (CPUs) y a la cantidad de memoria que posee el sistema de forma óptima, soportando de esta manera una mayor cantidad de peticiones simultáneas de forma correcta. Finalmente está diseñado para ambientes de alto volumen

1.10.3 SQLite

De la forma en que fue creado y diseñado SQLite, no necesita ser instalado, el mismo es un proyecto de dominio público, el cual implementa una pequeña biblioteca de aproximadamente 500kb, programado en el lenguaje C, totalmente libre, es un sistemas de bases de datos relacional. Además de ser independiente, simplemente se realizan llamadas a sub rutinas o

funciones de las propias librerías de SQLite, lo cual reduce ampliamente la latencia en cuanto al acceso a las bases de datos. (25)

Este posee una pequeña memoria y una única biblioteca necesaria para acceder a bases de datos incorporadas. Realiza operaciones de manera eficiente mucho más rápido que MySQL y PostgreSQL. Se ejecuta en muchas plataformas y sus bases de datos pueden ser fácilmente portadas sin ninguna configuración o administración. El mismo implementa un gran subconjunto de la ANSI – 92 SQL estándar, incluyendo sub-consultas, generación de usuarios, vistas y triggers. Cuenta con diferentes interfaces del API, las cuales permiten trabajar con C++, PHP, Perl, Python, Ruby, Tcl y groovy. Es de dominio público, y por tanto, se utiliza para cualquier propósito sin costo así como redistribuir libremente (25).

1.10.4 Selección del SGBD a utilizar

Teniendo en cuenta las características de almacenamiento que necesita el Laboratorio Virtual de Química, para su ejecución, se considera que la mejor opción es la de utilizar PostgreSQL, ya que reúne importantes ventajas como: es el SGBD de código abierto más avanzado del mundo, además es un potente servidor de Base de datos relacional orientado a objetos de uso libre, incluye extensa documentación y un enorme manual.

Consideraciones Parciales

Se realizó un estudio sobre las definiciones de los Laboratorios Virtuales, además de dar a conocer cuáles son los motivos para contar con PLVQ que respondan al programa de estudio de las secundarias básicas. Se efectuó un profundo análisis de algunas herramientas y tecnologías para la realización de la propuesta, definiéndose RUP como metodología que guíe el proceso de desarrollo, como herramienta para el modelado se utilizará Visual Paradigm for UML 8.0 apoyado en el lenguaje UML. Se precisó emplear como lenguaje de desarrollo C++ y el Gestor de Bases de Datos Postgres 8.4 y como herramienta de diseño gráfico Blender. Por último se optó QT Creator., como herramienta para el desarrollo de esta propuesta.

Capítulo 2: Características del sistema.

Introducción

Este capítulo constituye la propuesta de solución a través de la cual se realizará el LV de Química. Para la descripción de la solución se desarrolla el modelo del negocio para un mejor entendimiento del sistema. Se realiza la especificación de los requisitos funcionales y no funcionales que debe presentar el sistema. Además, se identifican y describen los casos de uso del sistema y sus actores. De manera general el capítulo recoge entre otros aspectos el flujo de trabajo requisito de la metodología RUP escogida en el capítulo anterior.

2.1 Modelo del negocio

El Modelado de Negocios se orienta a explicar cómo la empresa crea valor para el cliente. El modelo de negocios se define como una herramienta conceptual que contiene un conjunto de objetos, conceptos y sus relaciones con el objetivo de expresar la lógica del negocio de una empresa. Es una abstracción de cómo una empresa funciona, proporciona una vista simplificada de la estructura de negocios que actúa como la base para la comunicación, mejoras o innovación y define los requisitos de los sistemas de información que apoyan a la empresa (26)

2.2 Descripción del negocio actual

En la mayoría de las secundarias básicas de nuestro país, en la asignatura de Química, se necesita un paquete de prácticas de laboratorios que abarque los temas siguientes:

Las disoluciones acuosas de las sales. Concentración Másica.

La Separación de los componentes de las mezclas. Filtración.

La Separación de los componentes de las mezclas. Vaporización y destilación.

Estos temas antes expuestos permiten que se desarrollen en la asignatura habilidades básicas en los estudiantes y lograr un mejor entendimiento de los procesos químicos. Por la importancia que tiene el conocimiento y dominio de las características de la Química, es una propuesta lógica y necesaria que el estudiante de secundaria básica en la asignatura de Química, ejercite los conocimientos adquiridos en clases y experimente con sustancias haciendo uso de los instrumentos necesarios para el desarrollo de experimento. Actualmente, en la mayoría de los centros de enseñanza media del país, no se cuenta con instrumentos de

medición y de sustancias que son costosos, imposibilitando que el estudiante desarrolle determinadas habilidades relacionadas con la actividad experimental.

Una respuesta adecuada a esta exigencia de desarrollo de prácticas asociadas a la asignatura Química debe ser la automatización de las prácticas de laboratorio; permitiendo a los estudiantes de 8vo grado, experimentar con los instrumentos y químicos requeridos. A través de estas actividades los estudiantes podrán alcanzar los saberes para identificar y desarrollar la práctica, en particular la del campo químico. Facilitándole observar el resultado del experimento y analizar las acciones correctas e incorrectas al realizar la práctica de laboratorio.

2.3 Reglas del negocio

Las reglas de negocio describen políticas que deben cumplirse o condiciones que deben satisfacerse, por lo que regulan algún aspecto del negocio.

1. La práctica puede ser utilizada únicamente por los estudiantes y profesores de la secundaria.
2. El estudiante solo tendrá derecho a ver sus errores luego de haber realizado la práctica y no tendrá derechos de ver la de sus compañeros.
3. El estudiante solo tendrá derecho a ver sus resultados luego de haber realizado la práctica.
4. Al estudiante solo se le permitirá ver su nota una vez terminada la práctica.
5. El profesor es la única persona autorizada a ver las notas de sus estudiantes

2.4 Modelo de Casos de Uso del negocio

El modelo de casos de uso del negocio describe los procesos de negocio de una empresa en términos de casos de uso y actores del negocio, que se corresponden con los procesos del negocio y los clientes, respectivamente. El modelo de casos de uso se utiliza para conseguir un acuerdo con los usuarios y clientes sobre qué debería hacer el sistema para los usuarios, además de que nos ayuda a delimitar el sistema definiendo todo lo que debe hacer para sus usuarios. (8)

2.4.1 Actores del negocio

Un actor definirá qué roles pueden desempeñar los usuarios dentro del sistema y cuáles de estos roles pueden ser tomados conjuntamente por un mismo usuario.



Figura 1. Actor del negocio.

Actor	Descripción
Estudiante	Es el encargado de la realización de la práctica

Tabla 1. Descripción de Actores.

2.4.2 Caso de uso del negocio

Un caso de uso especifica una secuencia de acciones, incluyendo variantes, que el sistema puede llevar a cabo, y que producen un resultado observable de valor para un actor concreto.

(8)



Figura 2. Caso de uso del negocio.

2.4.3 Diagrama de casos de uso del negocio

Los diagramas de casos de uso del negocio constituyen una representación gráfica de un conjunto de elementos tales como actores y casos de uso, así como las relaciones y dependencias que se establecen entre ellos.

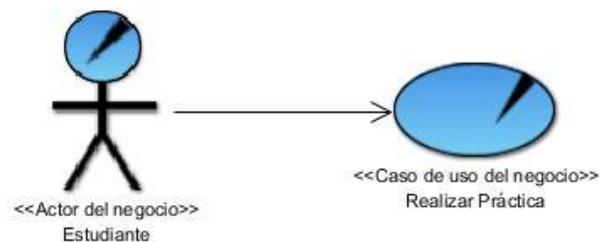


Figura 3. Diagrama de casos de uso del negocio.

2.4.4 Modelo de objetos del negocio

Es un modelo de objetos que describe cómo colaboran los trabajadores y las entidades del negocio dentro del flujo de trabajo del proceso de negocio.

2.4.5 Trabajador del negocio.

Un trabajador representa un puesto que puede ser asignado a una persona o a un grupo y especifica las responsabilidades y habilidades requeridas. Podemos decir que un trabajador representa una abstracción de un ser humano con ciertas capacidades que se requieren en un caso de uso del negocio (8)

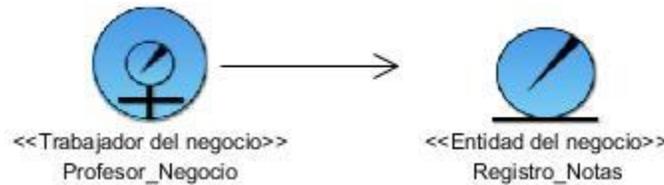


Figura 4. Diagrama de clases del modelo de objetos del negocio.

Trabajador del negocio	Descripción
Profesor	Es el encargado de orientarle al estudiante la práctica a realizar y emite la nota de la práctica

Tabla 2. Descripción de trabajadores del negocio.

2.4.6 Especificación de los casos de uso del negocio.

Caso de uso:	Realizar Práctica
Actor:	Estudiante
Propósito:	El sistema permite al estudiante realizar la prácticas
Resumen:	El caso de uso se inicia cuando el usuario procede a realizar la práctica, una vez terminado el profesor emitirá una nota y se le dará a conocer la respuesta.
Flujo Normal	
Acción del actor	Respuesta del proceso del negocio
1. Llega al laboratorio	1.1. El profesor orienta la práctica.
2. Realiza la práctica orientada	2.1 El profesor revisa la práctica. 2.2 Emite una nota

	2.3 El profesor actualiza el registro con la nota obtenida del estudiante
3. Recibe la nota emitida por el profesor.	
Flujo Alterno	
	2.1 Si el estudiante diera una respuesta incorrecta se le daría la correcta.

Tabla 3. Caso de Uso Realizar Práctica.

2.4.7 Diagrama de Actividades

Los Diagramas de Actividades ayudan a describir el detalle de qué es lo que pasa dentro del negocio, y para ello examinamos los roles específicos que juegan las personas (trabajadores del negocio) y las actividades que realizan. Ayudan a identificar que funciones deberá asumir el producto de software y quiénes serán los actores del futuro sistema.

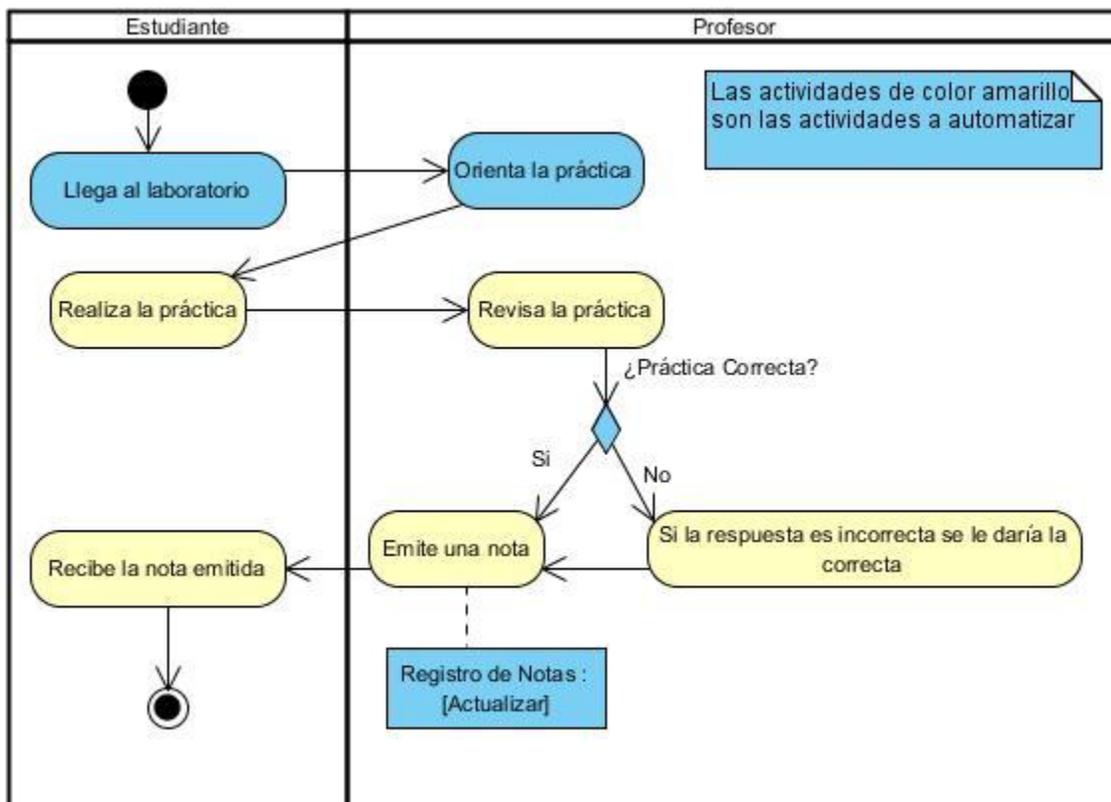


Figura 5. Diagrama de Actividades.

2.5 Requisitos.

Este es uno de los flujos de trabajo más importantes en el ciclo de vida del proyecto, porque en él se establece "qué" es lo que tiene que hacer exactamente el sistema. Los requisitos son un contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que se especifiquen.

2.5.1 Requisitos Funcionales.

Los requisitos funcionales establecen el comportamiento interno del sistema, son las funcionalidades específicas que muestran cómo se llevarán a cabo los casos de uso en la práctica.

RF 1 Registrar usuario:

El sistema permitirá al usuario registrarse. Este requisito cuenta con los siguientes campos: Nombre y Apellidos, CI, Contraseña, la opción Aceptar y Cancelar.

RF2 Autenticar usuario:

El sistema permitirá al usuario autenticarse. Este requisito cuenta con los siguientes campos: Nombre y Apellidos, Contraseña, la opción Aceptar y Cancelar.

RF3 Eliminar usuario:

El sistema permitirá al administrador eliminar usuarios en caso de que ya sea baja de la escuela. Este requisito cuenta con los siguientes campos: Categoría: donde se selecciona el tipo de usuario a eliminar en este caso (Estudiante, Profesor o Administrador), Lista clasificada por categoría: se muestra un listado con los usuarios de la categoría seleccionada y la opción Aceptar y Cerrar.

RF4 Asignar categoría:

El sistema permitirá al administrador asignar las categorías una vez que el usuario se autentique: Este requisito cuenta con los siguientes campos: Categoría: donde se selecciona el tipo de usuario al que desea asignar categoría en este caso (Estudiante, Profesor o Administrador), Listado de usuarios registrados sin categoría: son los usuarios que están registrados y que están en espera de que se le asigne la categoría, Lista clasificada por categoría: son los usuarios registrados que ya se le ha asignado la categoría , flecha: Pasar para la lista clasificada por categoría.

RF5 Crear laboratorio:

El sistema permitirá al usuario insertar todos los objetos gráficos según el laboratorio en que se encuentre. Estos laboratorios serán:

Tema: Las disoluciones acuosas de las sales. Concentración Másica.

Tema: Separación de los componentes de las mezclas. Filtración.

Tema: Separación de los componentes de las mezclas. Vaporización y destilación.

RF6 Mostrar escena:

El sistema deberá mostrar la escena con los objetos que contiene la escena después de cargada.

RF7 Administrar objeto:

El sistema deberá permitir trabajar con los objetos gráficos dentro de la escena.

RF7.1 Mover objeto:

El sistema permitirá al usuario mover los objetos dentro de la escena.

RF7.2 Eliminar objeto:

El sistema permitirá al usuario eliminar el objeto que no desee en la escena.

RF7.3 Arrastrar y soltar objeto:

La aplicación permite arrastrar y soltar el objeto encima de los otros objetos dentro del área de trabajo.

RF8 Devolver resultado de la práctica:

El sistema permitirá al usuario obtener los resultados de la práctica realizada, generando un pdf con los errores cometidos y la nota que se alcanzó al realizar la práctica, además de los datos del estudiante.

RF9 Guardar proyecto:

El sistema permitirá que el usuario pueda guardar la práctica de laboratorio.

RF10 Mostrar notas:

El sistema permitirá mostrar al profesor un listado con las notas de los estudiantes de su grupo que han realizado la práctica. Cuenta con los siguientes campos: Grupo, Listado de nota de los estudiantes: se mostrará el nombre del estudiante seguido de la nota, además de la opción Aceptar y Cancelar.

RF11 Conformar grupo docente:

El sistema le permitirá al profesor conformar los grupos docentes con el listado de los estudiantes registrados. Cuenta con los siguientes campos: grupo, lista de estudiantes sin grupo asociado: son los estudiantes que no están asignados a un grupo, lista de estudiantes de ese grupo: son los estudiantes que son de ese grupo entrado, un símbolo de flecha para ambos lados de tal manera permita de pasar de un lado a otro el estudiante seleccionado, Cerrar.

2.5.2 Requisitos no Funcionales.

Los requisitos no funcionales representan aquellos atributos que debe exhibir el sistema, pero que no son una funcionalidad específica.

RNF Usabilidad:

RNF 1. Permitir uso del teclado para realizar operaciones sobre el sistema.

RNF 2. El usuario podrá acceder a la práctica de laboratorio siempre que se encuentre registrado.

RNF 3. Finalidad: La aplicación tiene la finalidad de ofrecerle conocimientos al usuario a cerca de la realización de operaciones químicas mediante la interacción con la aplicación.

RNF Soporte:

RNF 1. Se debe brindar mantenimiento y actualización a la aplicación.

RNF Restricciones de diseño:

RNF 1. Se utiliza QT Creator como IDE de desarrollo.

RNF 2. En el trabajo con gráficos tridimensionales se utiliza la herramienta Blender.

RNF 3. Implementación del sistema. Se regirá por la filosofía de Programación Orientada a Objetos y se utilizará el motor gráfico Ogre.

RNF Apariencia o interfaz externa:

RNF 1. Tipo de Aplicación Informática: Aplicación de escritorio.

RNF 2. La interfaz de la aplicación estará formada por todos los elementos gráficos con los que contará la práctica de laboratorio virtual.

RNF Seguridad:

RNF 1. El usuario solo podrá acceder a la práctica de laboratorio siempre que se encuentre registrado.

RNF 2. El usuario debe autenticarse para poder acceder a la realización de la práctica de laboratorio.

RNF Hardware:

RNF 1. Instalación de la herramienta. Para la instalación del software la computadora destino deberá cumplir con los siguientes requisitos.

- 512 Mb de memoria RAM.
- Velocidad de Microprocesador a 3.0 GHz o superior.

RNF Software:

RNF 1. Un servidor de base de datos PostgreSQL.

RNF 2. El sistema deberá ser multiplataforma, es decir, deberá correr sobre los sistemas operativos Windows y Linux en las distribuciones Ubuntu, Debian, NOVA.

RNF Disponibilidad:

RNF 1. El sistema deberá estar disponible las 24 horas del día para todos los usuarios con derechos a utilizar el sistema.

RNF Requisitos de Licencia:

RNF 1. Se basan en herramientas con licencias GPL, se pueden instalar y utilizar en cualquier ordenador sin limitaciones.

2.6 Modelos de casos de uso del sistema

El modelo de casos de uso del sistema se basa en la descripción de elementos o usuarios externos al sistema (actores) y de la funcionalidad del sistema (casos de uso), además describe los requerimientos funcionales de un actor en términos de las interacciones que éste ejecuta con el sistema, es una técnica efectiva y a la vez simple para modelar los requerimientos del sistema desde la perspectiva del usuario. Presenta el sistema desde la perspectiva de su uso y esquematiza como proporcionará valor a sus usuarios.

2.6.1 Actores del Sistema.

Un actor definirá qué roles pueden desempeñar los usuarios dentro del sistema y cuáles de estos roles pueden ser tomados conjuntamente por un mismo usuario.

Estudiante: Encargado de resolver la práctica y recibe el reporte con la nota que obtuvo al realizar la misma, además es la generalización del profesor que inicializa las funcionalidades del estudiante además de él.

Profesor: Encargado de conformar los grupos docentes y ver la nota obtenida.

Administrador: Encargado de dar los privilegios una vez que el usuario se autentique en dependencia de si es un estudiante o profesor.

Usuario: Es la generalización de estudiante y administrador que inicializa la funcionalidad de autenticar usuario y registrar usuario

2.6.2 Casos de uso del sistema.

Los casos de uso del sistema nos ayudan a describir qué es lo que el sistema debe hacer. Los Casos de Uso son qué hace el sistema desde el punto de vista del usuario. Es decir, describen un uso del sistema y cómo este interactúa con el usuario, permiten acordar, entre el equipo de desarrollo y el cliente, los límites y los requisitos funcionales de dicho sistema.

2.6.3 Diagrama de Casos de Uso del Sistema.

Los diagramas de casos de uso sirven para facilitar la comunicación con los futuros usuarios del sistema, con el cliente y resultan especialmente útiles para determinar las características necesarias que tendrá el sistema. En otras palabras, los diagramas de casos de uso describen qué es lo que debe hacer el sistema, pero no cómo.

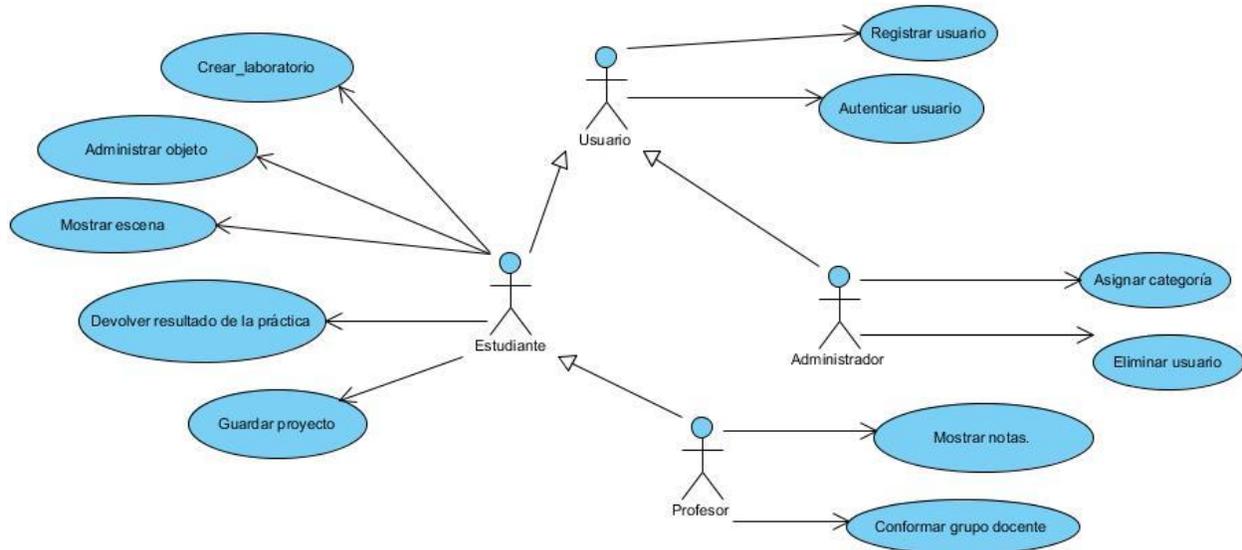


Figura 6. Diagrama de casos de uso del sistema.

2.6.4 Descripción de casos de uso.

Las descripciones de casos de uso son reseñas textuales del caso de uso. Normalmente tienen el formato de una nota o un documento relacionado de alguna manera con el caso de uso, y explica los procesos o actividades que tienen lugar en el caso de uso.

Caso de uso:	Registrar usuario.
Actor:	Usuario (Inicia).
Propósito:	El sistema permite al usuario registrarse.
Resumen:	El caso de uso se inicia cuando el usuario desea registrarse en la aplicación y finaliza cuando se le da cumplimiento al caso de uso. En caso contrario se notifica el fracaso mostrando un mensaje.
Referencia:	RF1
Precondiciones:	El usuario tiene que haber ejecutado la aplicación.
Poscondiciones:	Se registró un usuario en el sistema
Flujo Normal	
Acción del actor	Respuesta del sistema
1 Selecciona la opción registrarse.	1.1. Muestra en la ventana principal los siguiente campos: <ul style="list-style-type: none"> • Nombre y Apellidos. • CI. • Contraseña.
2 Llena los campos que se muestran en la ventana principal los cuales se encuentran en el siguiente formato: <ul style="list-style-type: none"> • Nombre y Apellidos. • CI. • Contraseña. Seleccionando por último la opción Aceptar.	2.1. Verifica que los campos estén llenos y tengan el formato correcto. 2.2. Validar Datos. 2.3. En caso de que los datos sean correctos se verifica que el usuario no exista en la base de datos. 8.1 2.4. Se registra satisfactoriamente permitiéndole al administrador asignarle categoría según el tipo de usuario si es estudiante o profesor, para que el usuario pueda pasar a autenticarse y pasar a la realización de la práctica.

Flujo Alterno	
Nº Evento 2.2. “Campos incompletos”	
Acción del actor	Respuesta del sistema
	2.2. Emitirá un mensaje “Datos incorrectos”. Volver al paso 1.1.
Nº Evento 2.3. “El usuario existe”	
Acción del actor	Respuesta del sistema
	2.3. Mostrara un mensaje “El usuario ya existe”. Volver al paso 1.1.
Prioridad:	Crítico.
Complejidad:	Alta.

Tabla 4. CUS 1 Registrar usuario.

Caso de uso:	Autenticar usuario.
Actor:	Usuario (Inicia).
Propósito:	El sistema permite al usuario autenticarse en el sistema.
Resumen:	El caso de uso se inicia cuando el usuario accede a su interfaz de usuario y finaliza cuando el usuario accede al sistema. En caso contrario se notifica el fracaso mostrando un mensaje.
Referencia:	RF2
Precondiciones:	El usuario tiene que haberse registrado.
Poscondiciones:	El usuario pudo acceder a la interfaz de usuario.
Flujo Normal	
Acción del actor	Respuesta del sistema
1. Selecciona la opción autenticarse.	1.1. Muestra en la ventana principal los siguiente campos: <ul style="list-style-type: none"> • Nombre y Apellidos. • Contraseña.

Capítulo 2: Característica del Sistema

2. El usuario accede a autenticarse llenando los campos que se encuentran en la ventana principal nombre y apellidos, contraseña y envía la información presionando el botón "Aceptar".	2.1. Verifica usuario y contraseña. 1.1. Verifica que el usuario es del dominio uci.cu. 1.2. En caso de que los datos sean correctos se le asignan los permisos para acceder a la práctica y privilegios según el tipo de usuario si es estudiante o profesor. 2.2. Muestra la lista clasificada por categoría.
Flujo Alterno	
Acción del actor	Respuesta del sistema
	2.1 Los datos son incorrectos el sistema emite un mensaje "Datos incorrectos". Volver al paso 1.1.
Prioridad:	Crítico.
Complejidad:	Alta.

Tabla 5. CUS 2 Autenticar usuario.

Caso de uso:	Eliminar usuario.
Actor:	Administrador (Inicia).
Propósito:	El sistema permitirá al administrador eliminar usuarios
Resumen:	El caso de uso se inicia cuando el administrador elimina a los usuarios que ya sean baja de la escuela.
Referencia:	RF3
Precondiciones:	El administrador tiene que estar autenticado.
Poscondiciones:	El administrador elimina el usuario.
Flujo Normal	
Acción del actor	Respuesta del sistema
1. Selecciona la opción "Eliminar usuario".	1.1. Muestra usuario y contraseña siguiente verifica que el usuario es del dominio uci.cu. 1.2. En caso de que los datos sean correctos se le asignan los permisos para acceder a la práctica y privilegios según el tipo de usuario si es estudiante o profesor. • Muestra la lista clasificada por categoría.
2. Selecciona si es a un estudiante, profesor o administrador que desea eliminar.	2.1. Muestra el listado clasificado por categoría según la selección del administrador.
3. Selecciona el usuario en la lista clasificada por categoría y presiona el botón "Eliminar".	3.1. Elimina el usuario seleccionado de la lista clasificada.

Flujo Alterno	
Acción del actor	Respuesta del sistema
3. No selecciona el usuario de la lista clasificada por categoría y presiona el botón “Eliminar”.	3.1 Emitirá un mensaje “Debe seleccionar el usuario de la lista clasificada”. Volver al paso 2.1
Prioridad:	Crítico.
Complejidad:	Media.

Tabla 6. CUS 3 Eliminar usuario

Caso de uso:	Asignar categoría.
Actor:	Administrador (Inicia).
Propósito:	El sistema permitirá al administrador asignar categoría.
Resumen:	El caso de uso se inicia cuando el administrador pasa a asignar categoría a los usuarios que se encuentran registrados en dependencia de si es un estudiante o profesor y finaliza cuando el administrador ha asignado la categoría. En caso contrario se notifica el fracaso mostrando un mensaje.
Referencia:	RF4
Precondiciones:	El administrador tiene que estar autenticado, además de que los usuarios tienen que estar registrados para poder asignar categoría.
Poscondiciones:	El administrador pudo asignar categoría.
Flujo Normal	
Acción del actor	Respuesta del sistema
1. Selecciona la opción “Asignar Categoría”.	<p>1.1. Busca usuario y compara la contraseña siguiente formato</p> <p>1.2. En caso de que los datos sean correctos se le asignan los permisos para acceder a la página y privilegios según el tipo de usuario si es estudiante o profesor.</p> <ul style="list-style-type: none"> Listado de usuarios registrados sin categoría. Lista clasificada por categoría.

Capítulo 2: Característica del Sistema

2. Selecciona la categoría (estudiante, profesor o administrador) que desea asignar al usuario. Selecciona de la lista de usuarios registrados sin categoría, el usuario deseado y lo clasifica según la categoría.	2.1. Asigna la categoría y selecciona la contraseña. verifica que el usuario es del dominio uci.cu. 1.2. Pasa de la lista de usuarios registrados sin categoría a la lista clasificada por categoría, privilegios según el tipo de usuario si es estudiante o profesor.
Flujo Alterno	
Acción del actor	Respuesta del sistema
2. Si el usuario asigna por equivocación una categoría, podrá quitar dicha categoría seleccionando de la "lista clasificada por categoría" al usuario que se le fue asignado por error y selecciona la flecha "Pasar para la lista de usuarios registrados sin categoría".	2.1 El sistema le muestra la contraseña para la verifica que el usuario es del dominio uci.cu. 1.2 "Lista de usuarios registrados sin categoría" En caso de que los datos sean correctos se le 2.2. Se le muestra la flecha para acceder a la práctica y privilegios según el tipo de usuario si es estudiante o profesor.
Prioridad:	Crítico.
Complejidad:	Alta.

Tabla 7. CUS 4 Asignar categoría.

Caso de uso:	Crear laboratorio.
Actor:	Estudiante (Inicia).
Propósito:	El sistema permite al estudiante crear el laboratorio.
Resumen:	El caso de uso se inicia cuando el estudiante selecciona cual de las 3 prácticas desea desarrollar.
Referencia:	RF5
Precondiciones:	El usuario tiene que haberse autenticado para acceder a la práctica.
Poscondiciones:	El estudiante creó la escena práctica.
Flujo Normal	
Acción del actor	Respuesta del sistema
1. Selecciona cual de las 3 prácticas desea realizar presionando el botón con el nombre de la misma.	1.1 Muestra el área de trabajo de la práctica. 1.2 Muestra un panel de herramientas donde se encuentran todos los objetos gráficos. Estos objetos serán, para el laboratorio1-Escenario1:

Capítulo 2: Característica del Sistema

Tema: Las disoluciones acuosas de las sales.

Concentración Másica

- Vaso de precipitados.
- Vidrio reloj (Solute de sulfato de cobre 2 (CuSO₂)).
- Frasco lavador (agua destilada).
- Agitador de vidrio.
- Embudo liso.
- Vaso de precipitados más pequeño.
- Pipeta.

El laboratorio2-Escenario2:

Tema: Separación de los componentes de las mezclas.

Filtración.

Herramientas que se utilizan:

- Soporte universal aro.
- Mordaza.
- Vasos de precipitados.
- Agitador.
- Embudo liso.
- Papel de filtro.

El laboratorio3-Escenario3:

Separación de los componentes de las mezclas.

Vaporización y destilación.

- Vidrio reloj (Cloruro de sodio).
- Vasos de precipitados (agua).
- Agitador.
- Cápsula de porcelana.
- Soporte universal.
- Mechero de alcohol.
- Pinza de crisol.
- Aro.
- Mordaza.

2. Selecciona el objeto necesario para la realización de la práctica.	2.1 Posiciona el objeto seleccionado por el estudiante encima de la mesa que se encuentra en la escena.
Prioridad:	Crítico.
Complejidad:	Media.

Tabla 8. CUS 5 Crear laboratorio.

Caso de uso	Mostrar escena.	
Actor:	Estudiante (Inicia).	
Propósito	El sistema deberá permitir cargar la escena anteriormente guardada, mostrando los objetos de la práctica que contienen la misma.	
Resumen:	El caso de uso se inicia cuando el estudiante carga la escena donde ha realizado la práctica de laboratorio.	
Referencia	RF6	
Precondiciones	El usuario tiene que haberse autenticado para acceder a la práctica.	
Poscondiciones	Todos los objetos correspondientes a la práctica de laboratorio fueron mostrados.	
Flujo Normal		
Acción del actor	Respuesta del sistema	
1. Selecciona la opción “Abrir Proyecto”.	1.1 Muestra una ventana para localizar la ruta donde se encuentra la escena.	
2. Selecciona la escena y presiona el botón “Aceptar”.	2.1 Carga la escena mostrando los objetos que se encuentran en la misma.	
Flujo Alternativo		
Acción del Actor	Respuesta del Sistema	
2. Selecciona la opción “Cancelar”.	2.1. Cierra la ventana.	
Prioridad:	Secundario.	
Complejidad:	Media.	

Tabla 9. CUS 6 Mostrar la escena guardada.

Caso de uso	Administrar objeto.
Actor:	Estudiante (Inicia).

Propósito	Permite al estudiante mover, eliminar, arrastrar y soltar el objeto sobre los otros objetos.
Resumen:	El caso de uso se inicia cuando el estudiante procede a desarrollar la práctica de laboratorio y finaliza cuando termina la misma.
Referencia	RF7
Precondiciones	En la escena tiene que haber algún objeto insertado.
Poscondiciones	El estudiante movió, eliminó, además de arrastrar y soltar el objeto.
Flujo Normal	
Acción del actor	Respuesta del sistema
1. Accede a la realización de la práctica.	
Sección “Mover objeto”	
2. Selecciona el objeto a mover.	2.1. Verifica que el objeto haya sido seleccionado por el estudiante. 2.2. Mueve el objeto seleccionado.
Sección “Eliminar objeto”	
3. Selecciona el objeto que desea eliminar.	3.1. Verifica que se haya seleccionado un objeto.3.1
4. Selecciona dentro de la opción “Editar”, la posibilidad de “Eliminar”.	4.1. Elimina el objeto seleccionado.
Sección “arrastrar y soltar objeto”	
5. Selecciona el objeto que desea arrastrar y soltar.	5.1. Verifica que el objeto haya sido seleccionado por el estudiante. 5.2. Permite arrastrar y soltar el objeto sobre otro.
Flujo Alternativo	
Nº Evento 2.1. “Mover objeto”	
Acción del actor	Respuesta del sistema
2. Selecciona el objeto.	2.1. Verifica que el objeto haya sido seleccionado por el estudiante. Muestra mensaje “Debe seleccionar un objeto”. Volver al paso 2.
Nº Evento 3.1. “Eliminar objeto”	
Acción del actor	Respuesta del sistema

Capítulo 2: Característica del Sistema

3. Selecciona el objeto.	3.1. Verifica que se haya seleccionado un objeto. Muestra mensaje “Debe seleccionar un objeto”. Volver al paso 3
Prioridad:	Crítico.
Complejidad:	Alta.

Tabla 10. CUS 7 Administrar objetos.

Caso de uso	Devolver resultado de la práctica.	
Actor:	Estudiante (Inicia).	
Propósito	Permite al estudiante ver la nota obtenida en la realización de la práctica.	
Resumen:	El caso de uso se inicia cuando el estudiante desea generar el reporte de la práctica.	
Referencia	RF8	
Precondiciones	El estudiante tiene que haber terminado la práctica de laboratorio.	
Poscondiciones	Se muestra el resultado de la práctica.	
Flujo Normal		
Acción del actor	Respuesta del sistema	
1. Accede a la realización de la práctica.	1.1. Muestra al estudiante el área de trabajo.	
2. Selecciona la opción “Ver”, “Devolver resultado de la práctica realizada”.	2.1. Muestra una ventana en la cual el estudiante debe guardar el archivo pdf que generó.	
3. Selecciona la ruta donde desea guardar y presiona el botón “aceptar”.	3.1. Crea el documento pdf con los errores cometidos, la nota obtenida en la misma y datos del estudiante, guardándolo en la ruta seleccionada por el estudiante.	
Flujo Alterno		
Acción del actor	Respuesta del sistema	
3. El estudiante selecciona la opción “Cancelar”.	3.1. El sistema cierra la ventana.	
Prioridad:	Crítico.	
Complejidad:	Alta.	

Tabla 11. CUS 8 Devolver resultado de la práctica realizada.

Caso de uso	Guardar proyecto.	
Actor:	Estudiante (Inicia).	
Propósito	Permite al estudiante guardar la práctica realizada.	
Resumen:	El caso de uso se inicia cuando el estudiante desea guardar la práctica y termina una vez guardada.	
Referencia	RF9	
Precondiciones	El estudiante tiene que haberse autenticado.	
Poscondiciones	El estudiante pudo guardar la práctica.	
Flujo Normal		
Acción del actor	Respuesta del sistema	
1. Una vez dentro de la práctica accede a la opción "Guardar Archivo" que se encuentra en su área de trabajo.	1.1. Muestra una ventana en la cual el estudiante debe especificar donde desea guardarlo.	
2. Selecciona la ruta donde desea guardar y presiona el botón "aceptar".	2.1. Guarda la práctica en la ruta seleccionada por el estudiante.	
Flujo Alternativo		
Acción del actor	Respuesta del sistema	
2. Selecciona la opción "Cancelar".	2.1. Cierra la ventana.	
Prioridad:	Secundario.	
Complejidad:	Media.	

Tabla 12. CUS 9 Guardar proyecto.

Caso de uso	Mostrar notas.
Actor:	Profesor (Inicia).
Propósito	Permite ver al profesor las notas de los estudiantes.
Resumen:	El caso de uso inicia cuando el profesor accede al sistema para ver las notas que obtuvieron los estudiantes en la realización de la práctica.

Referencia	RF11
Precondiciones	El profesor debe haberse autenticado.
Poscondiciones	El profesor pudo ver las notas obtenidas por los estudiantes en la práctica.
Flujo Normal	
Acción del actor	Respuesta del sistema
1. Selecciona la opción “Mostrar notas”.	1.1. Le muestra una ventana con el siguiente formato: <ul style="list-style-type: none"> • Grupo. • Listado de estudiante. • Historial del estudiante seleccionado.
2. Selecciona el grupo para saber los estudiantes que pertenecer a ese grupo.	2.1. Muestra un listado de estudiantes del grupo seleccionado por el profesor.
3. Selecciona al estudiante del “listado de estudiante” para ver la nota obtenida en las diferentes prácticas.	3.1. Muestra el historial del estudiante seleccionado.
Prioridad:	Crítico.
Complejidad:	Media.

Tabla 13. CUS 11 Mostrar notas.

Caso de uso	Conformar grupo docente.
Actor:	Profesor (Inicia).
Propósito	Permite al profesor conformar los grupos docentes.
Resumen:	El caso de uso inicia cuando el profesor accede al sistema para conformar los grupos docentes con el listado de estudiantes registrados.
Referencia	RF12
Precondiciones	El profesor debe haberse autenticado.
Poscondiciones	Fue creado un nuevo grupo en el sistema.
Flujo Normal	
Acción del actor	Respuesta del sistema
1. Selecciona la opción “Conformar grupo docente”.	1.1. Le muestra una ventana con el siguiente formato: <ul style="list-style-type: none"> • Grupo.

	<ul style="list-style-type: none"> Listado de estudiantes sin grupo asociado. Listado de estudiantes de ese grupo.
2. Introduce el grupo y selecciona el estudiante de la lista de estudiante sin grupo asociado y lo pasa para la lista de estudiante de ese grupo entrado.	2.1. Pasa para la lista de estudiante de ese grupo los estudiantes seleccionados por el profesor.
Flujo Alterno	
Acción del actor	Respuesta del sistema
<p>2. No introduce el grupo.</p> <p>2. No seleccionó el estudiante de la lista de estudiante.</p> <p>2. Si el profesor asigna por equivocación un estudiante a un grupo al que no pertenece, podrá quitarlo de dicho grupo seleccionando de la lista de estudiante de ese grupo al usuario que se le fue asignado por error y selecciona la flecha pasar a la lista de estudiante sin grupo asociado.</p>	<p>2.1 Muestra mensaje “Debe introducir el grupo”.</p> <p>2.1 Muestra mensaje ”Debe seleccionar el estudiante”.</p> <p>2.1.El sistema permite pasar al usuario para la “Listado de estudiantes sin grupo asociado” Termina así el caso de uso.</p>
Prioridad:	Crítico.
Complejidad:	Media

Tabla 14. CUS 12 Conformar grupo docente.

Consideraciones Parciales

En este capítulo se mostraron las principales modelos del negocio y se definieron los actores que interactúan con el sistema, además de analizar las características y funciones fundamentales del sistema para el desarrollo de la propuesta del LV Estos se representaron mediante un diagrama de casos de uso, llegando así a las descripciones detalladas de los mismos. Una vez realizado el análisis se procede al diseño de un sistema automatizado, partiendo de los requisitos funcionales y no funcionales, que permitirán un mejor entendimiento del sistema, proporcionando la posibilidad de pasar a la fase de implementación del LV de Química.

Capítulo 3: Análisis y Diseño del Sistema

Introducción

En el presente capítulo se muestra el análisis y diseño para la solución de la aplicación, modelándose los artefactos necesarios que sirven como gráfico a la implementación, además del análisis de los patrones a utilizar. Se hace un estudio detallado del modelo de caso de uso del sistema con el propósito de diseñar las clases a implementar a través del Análisis, las que finalmente se representarán en un diagrama de clases del Diseño.

3.1 Patrones de Arquitectura

Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones. En comparación con los patrones de diseño, los patrones arquitectónicos tienen un nivel de abstracción mayor.

3.2 Patrón Modelo-Vista-Controlador (MVC):

MVC es un patrón de diseño de arquitectura de software usado principalmente en aplicaciones que manejan gran cantidad de datos y transacciones complejas donde se requiere una mejor separación de conceptos para que el desarrollo esté estructurado de una mejor manera, facilitando la programación en diferentes capas de manera paralela e independiente. MVC sugiere la separación del software en 3 componentes: Modelo, Vista y Controlador, los cuales serán explicados en breve:

Modelo: Es la representación de la información que maneja la aplicación. El modelo en sí son los datos puros que puestos en contexto del sistema proveen de información al usuario o a la aplicación.

Vista: Es la encargada de mostrar la información e interactuar con el usuario.

Controlador: Es la capa encargada de manejar y responder las solicitudes del usuario, procesando la información necesaria y modificando el Modelo en caso de ser necesario. Patrones de Diseño.

El primer objetivo de los patrones de diseño es el de capturar buenas prácticas que nos permitan mejorar la calidad del diseño de un sistema, determinando elementos que soporten roles útiles en dicho contexto, encapsulando complejidad, y haciéndolo más flexible.

3.2.1 Patrones de diseño aplicados

Los patrones GRASP describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones.

Experto: El GRASP de Experto en información es el principio básico de asignación de responsabilidades. Indica que la responsabilidad de la creación de un objeto debe recaer sobre la clase que conoce toda la información necesaria para crearlo, lo cual permite que se conserve el encapsulamiento, soportando un bajo acoplamiento y una alta cohesión.

Este patrón se manifiesta en la clase **Escena** que es la experta en la información relacionada con la escena y los objetos 3D.

Controlador: Este patrón funciona como intermediario entre una determinada interfaz y el algoritmo que la implementa, de tal forma que es la que recibe los datos y la que los envía a las distintas clases según el método llamado. (27)

El mismo se evidencia en la clase **CC_Gestionar_Usuario** y **CC_Práctica_Laboratorio**, pues funcionan como intermediarias entre la interfaz y el modelo contribuyendo a la reutilización de la lógica en futuras actualizaciones. Puesto que la lógica no está ligada a la capa de interfaz, puede sustituirse por una interfaz nueva.

Alta Cohesión: Asigna responsabilidades procurando que la cohesión sea lo más alta posible, es decir cada elemento del diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos.

Bajo Acoplamiento: Asigna responsabilidades de tal manera que el acoplamiento sea el menor posible es decir debe haber pocas dependencias entre las clases.

Estos dos últimos patrones se aplicaron en todas las clases del sistema tratando de incrementar la claridad y facilitando su comprensión, de manera tal que se simplifique el mantenimiento y las mejoras en el caso de la alta cohesión. Por su parte el bajo acoplamiento permitirá realizar cambios sin afectar a otras clases propiciando su reutilización.

De los patrones GoF se aplicó al diseño de clases, el siguiente:

Singleton: Es un patrón de tipo creacional, restringe la instanciación de una clase o valor de un tipo a un solo objeto. Se aplica este patrón a la solución del problema porque es necesario acceder desde varias clases a una misma. Especializar dicha clase fue la solución al problema, de esa forma se aseguraría que solo se cree una instancia de la clase, proporcionando un punto de acceso global a ella. Todos los objetos que utilizan una instancia de esa clase usan la misma instancia.

3.3 Análisis y Diseño

El objetivo del flujo Análisis y Diseño consiste en traducir los requisitos a una especificación que describe cómo implementar el sistema.

3.3.1 Diagrama de Clases del Análisis

Los diagramas de clases del análisis representan la relación entre las clases que intervienen en los casos de uso. Proporcionan una vista interna del sistema. Es utilizado para comprender de forma general la estructura del sistema y sirve como entrada para la etapa de diseño.

En el diagrama de clases del análisis podemos encontrar distintos tipos de clases como:

Entidad: Modelan información que posee larga vida y que es a menudo persistente.

Interfaz: Modelan la interacción entre el sistema y sus actores.

Controladora: Coordinan la realización de uno o unos pocos casos de uso coordinando las actividades de los objetos que implementan la funcionalidad del caso de uso.

El propósito del análisis es definir todas las clases que son relevantes al problema que se va a resolver, las operaciones y atributos asociados, las relaciones y comportamientos asociadas con ellas. (28)

Este modelo es usado para representar la estructura global del sistema, describe la realización de casos de uso y sirve como una abstracción del Modelo de Diseño. El Modelo de Análisis puede contener: las clases y paquetes de análisis, las realizaciones de los casos de uso, las relaciones y los diagramas. “Durante el análisis, se analizaron los requisitos que se describen

en la captura de requerimientos, refinándolos y estructurándolos. El objetivo de hacerlo es conseguir una comprensión más precisa de los requisitos y una descripción de los mismos que sea fácil de mantener y que nos ayude a estructurar el sistema entero, incluyendo su arquitectura”. (29)

A continuación se muestra la descripción del CU “Registrar usuario, Autenticar usuario, Asignar categoría y Crear laboratorio”. ([Ver Anexo I](#) para los restantes casos de uso).



Figura 7. Diagrama de clases del análisis Registrar usuario.

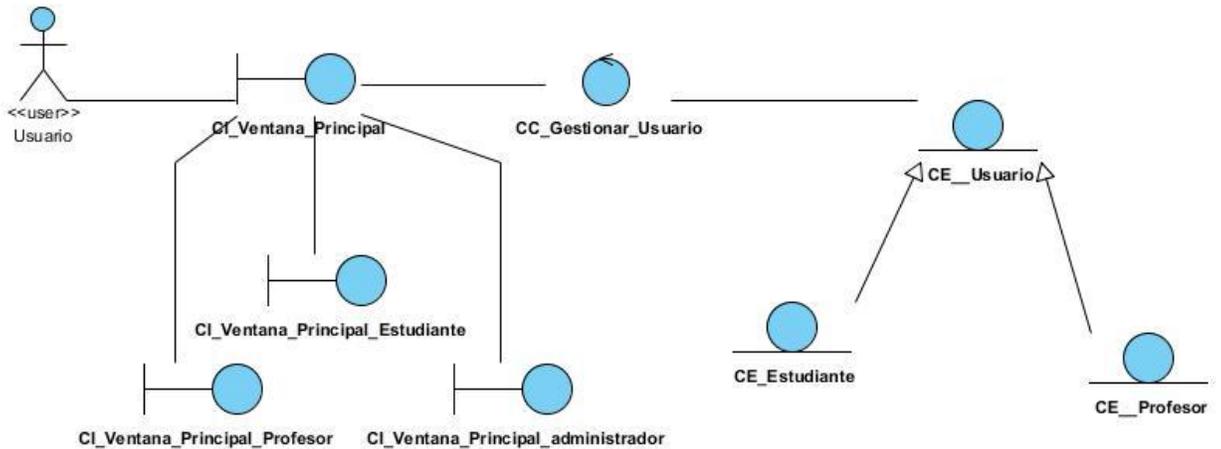


Figura 8. Diagrama de clases del análisis Autenticar usuario

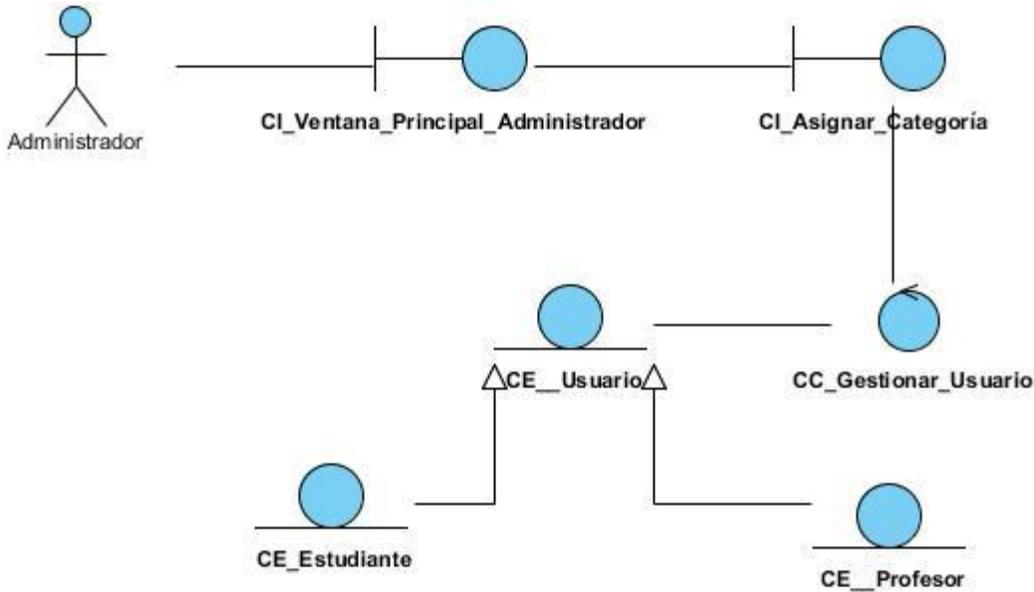


Figura 9. Diagrama de clases del análisis Asignar categoría

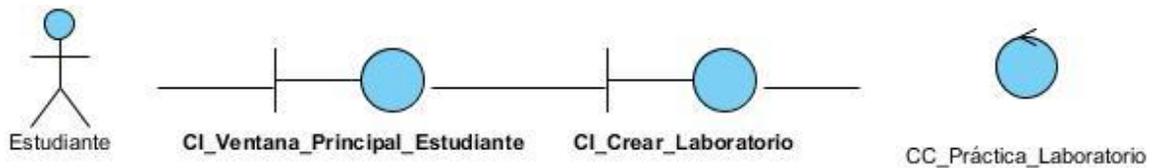


Figura 10. Diagrama de clases del análisis Crear laboratorio

3.3.2 Diagrama de Interacción

Los diagramas de interacción se utilizan para modelar los aspectos dinámicos de un sistema, lo que conlleva 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. En el contexto de las clases describen la forma en que grupos de objetos colaboran para proveer un comportamiento. Se dividen en dos categorías los diagramas de colaboración y los diagramas de secuencia.

3.3.2.1. Diagramas de Colaboración

Un diagrama de colaboración es un diagrama de interacción 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 muestra la descripción del CU “Registrar usuario, Autenticar usuario, Asignar categoría y Crear laboratorio”. ([Ver Anexo II](#) para los restantes casos de uso).

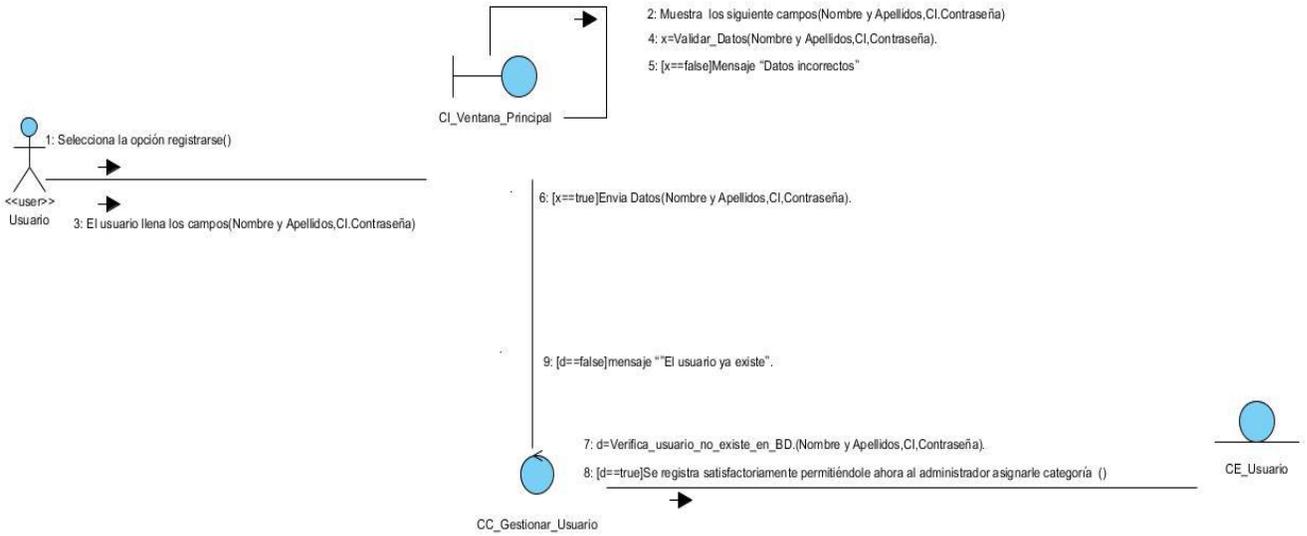


Figura 11. Diagrama de colaboración Registrar usuario.

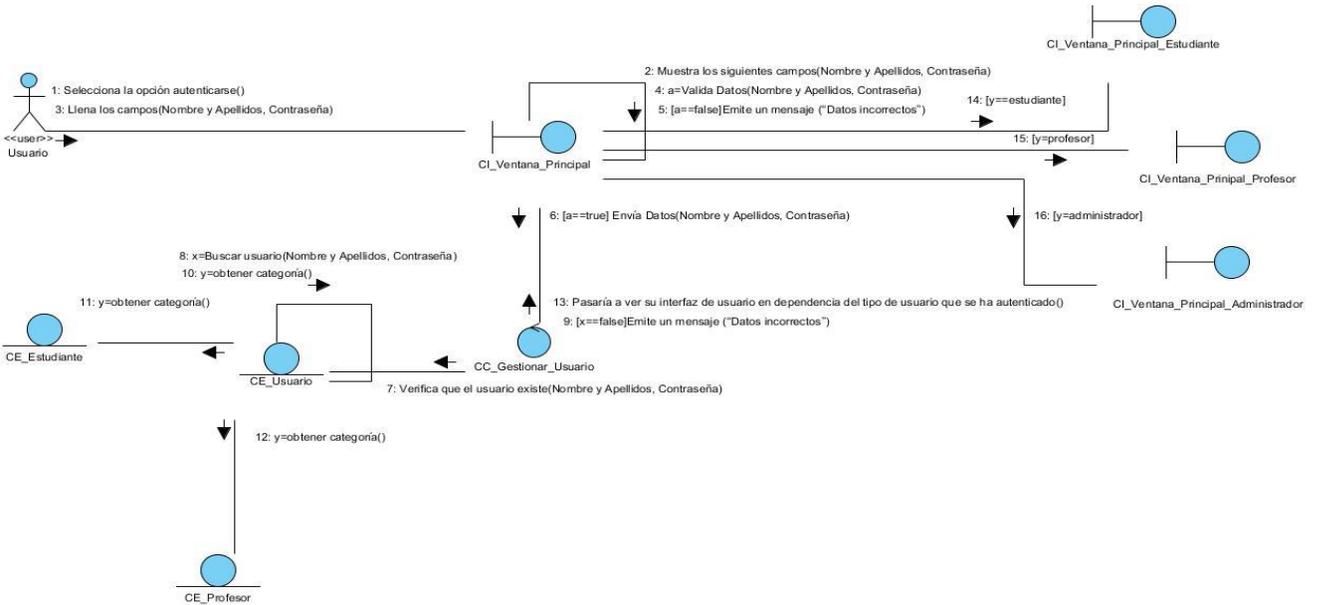


Figura 12. Diagrama de colaboración Autenticar usuario.

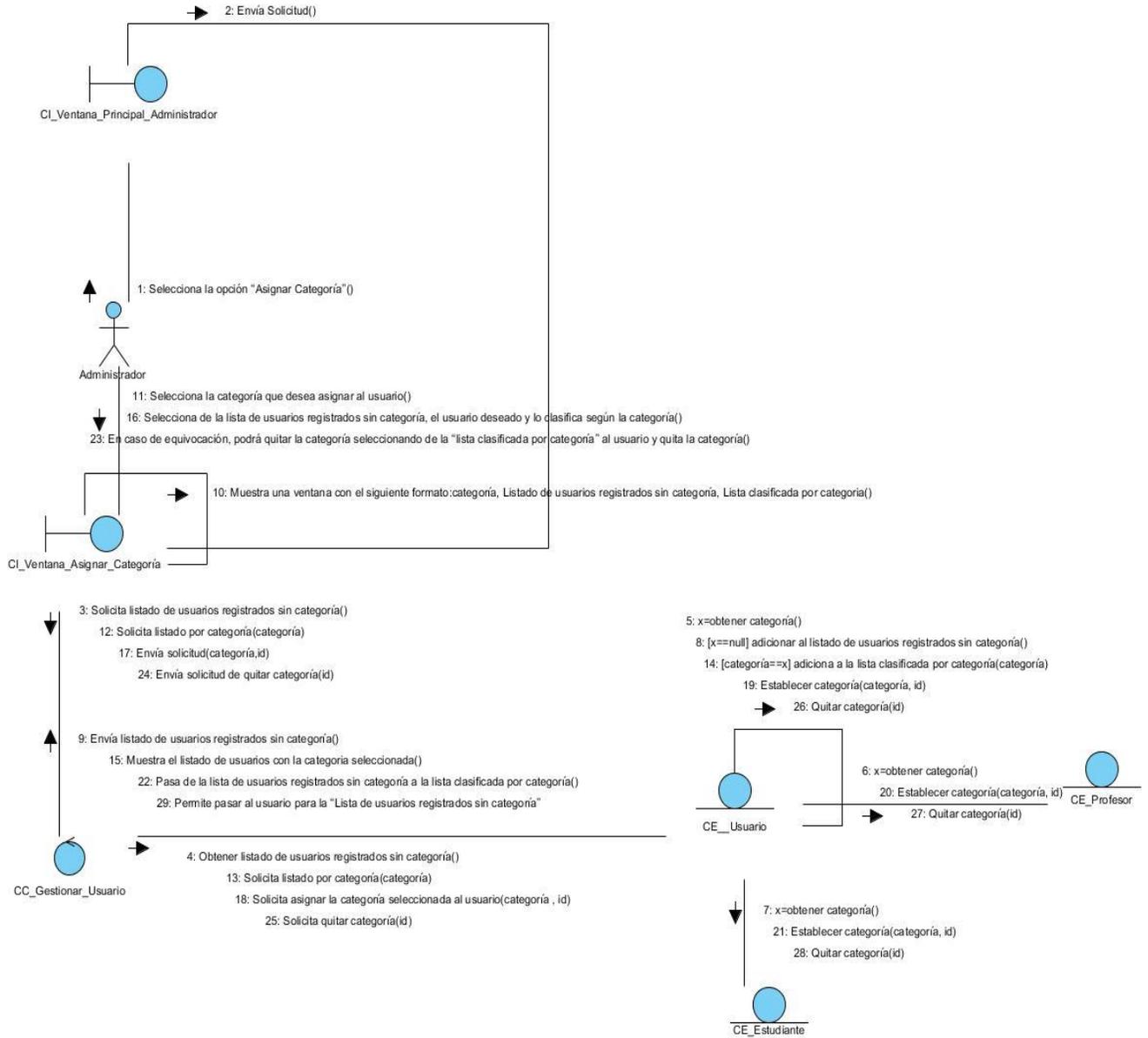


Figura 13. Diagrama de colaboración Asignar categoría.

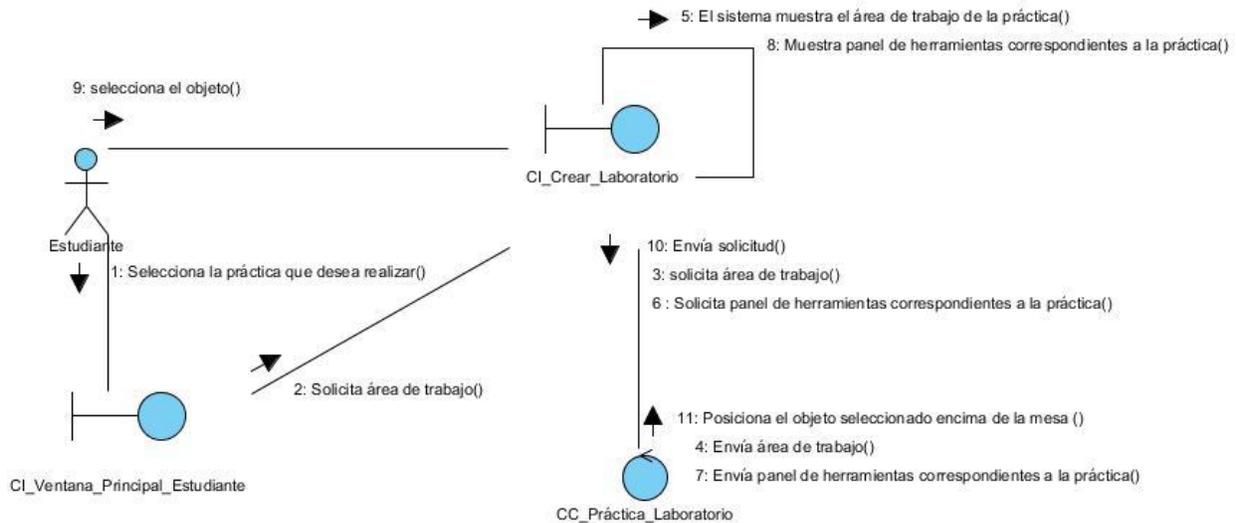


Figura 14. Diagrama de colaboración Crear laboratorio.

3.3.3 Diagrama de Clases del Diseño.

A través del flujo de diseño, uno de los artefactos más importantes a obtener son los diagramas de clases de diseño, donde se exponen las clases que intervienen en las realizaciones de los casos de uso del sistema. En este tipo de diagrama se representa un nivel de detalle más alto que los diagramas de clases del análisis, relacionándose con el lenguaje de programación del cual se hará uso en la implementación del sistema. El diagrama de clases de diseño describe gráficamente las especificaciones de las clases de software y de las interfaces en una aplicación. Normalmente contiene la siguiente información:

- Clases, asociaciones y atributos.
- Interfaces, con sus operaciones y constantes.
- Métodos
- Información sobre los tipos de los atributos, navegabilidad y dependencias

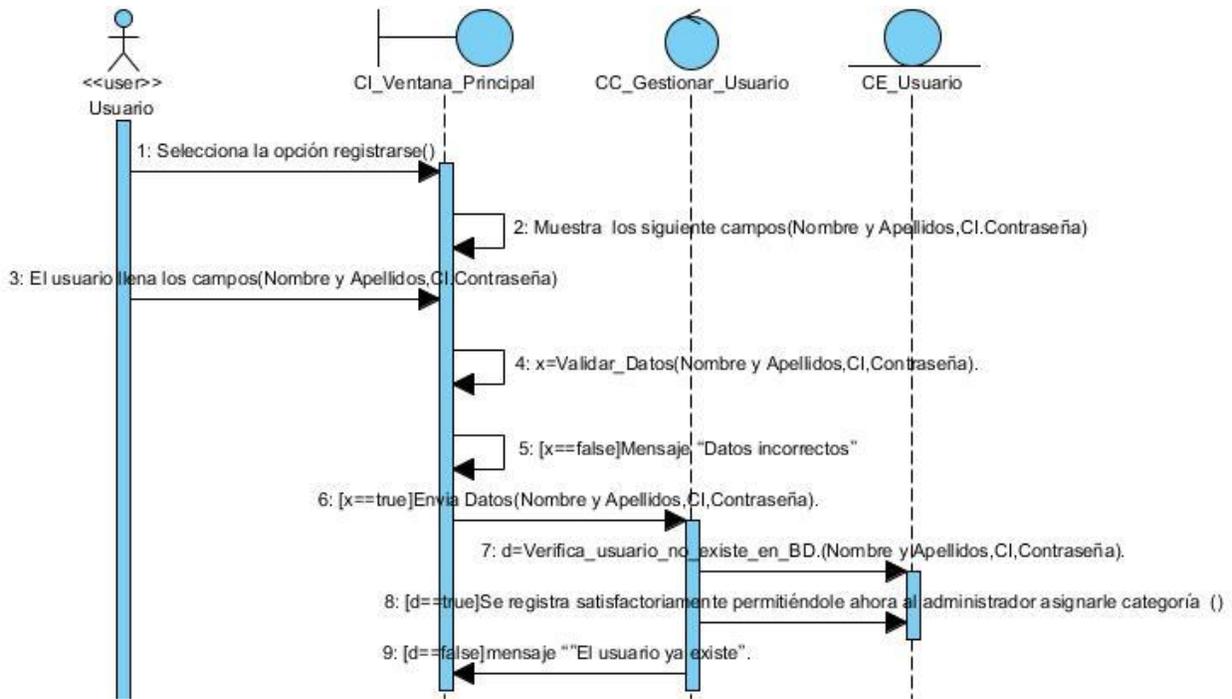


Figura 16. Diagrama de secuencia Registrar usuario.

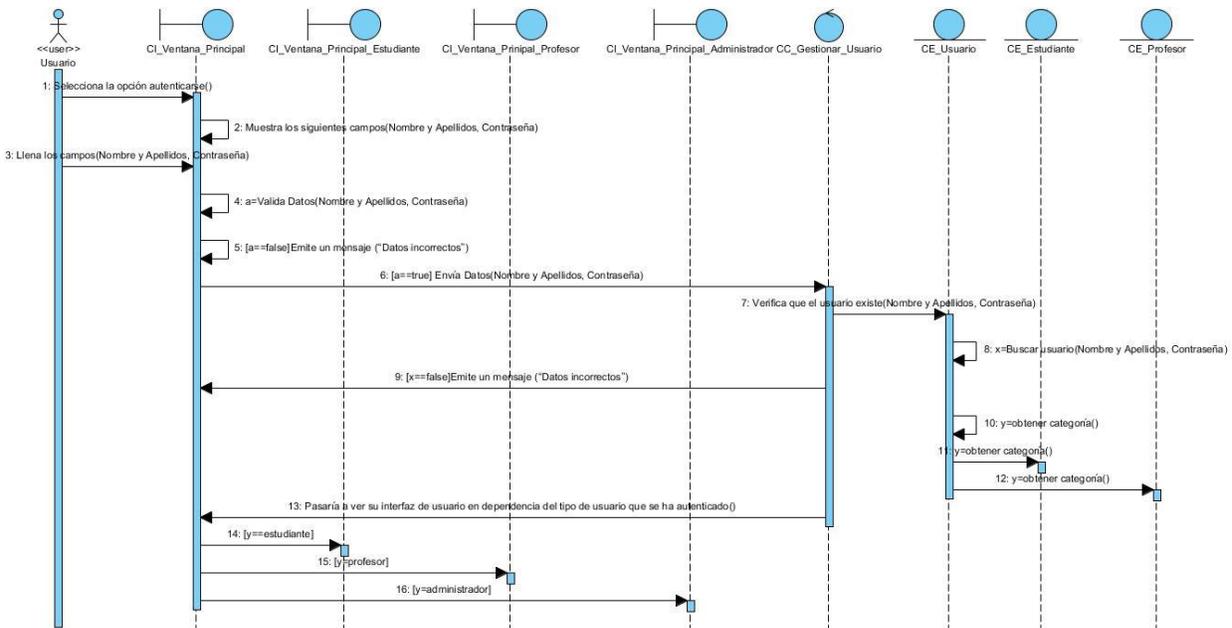


Figura 17. Diagrama de secuencia Autenticar usuario.

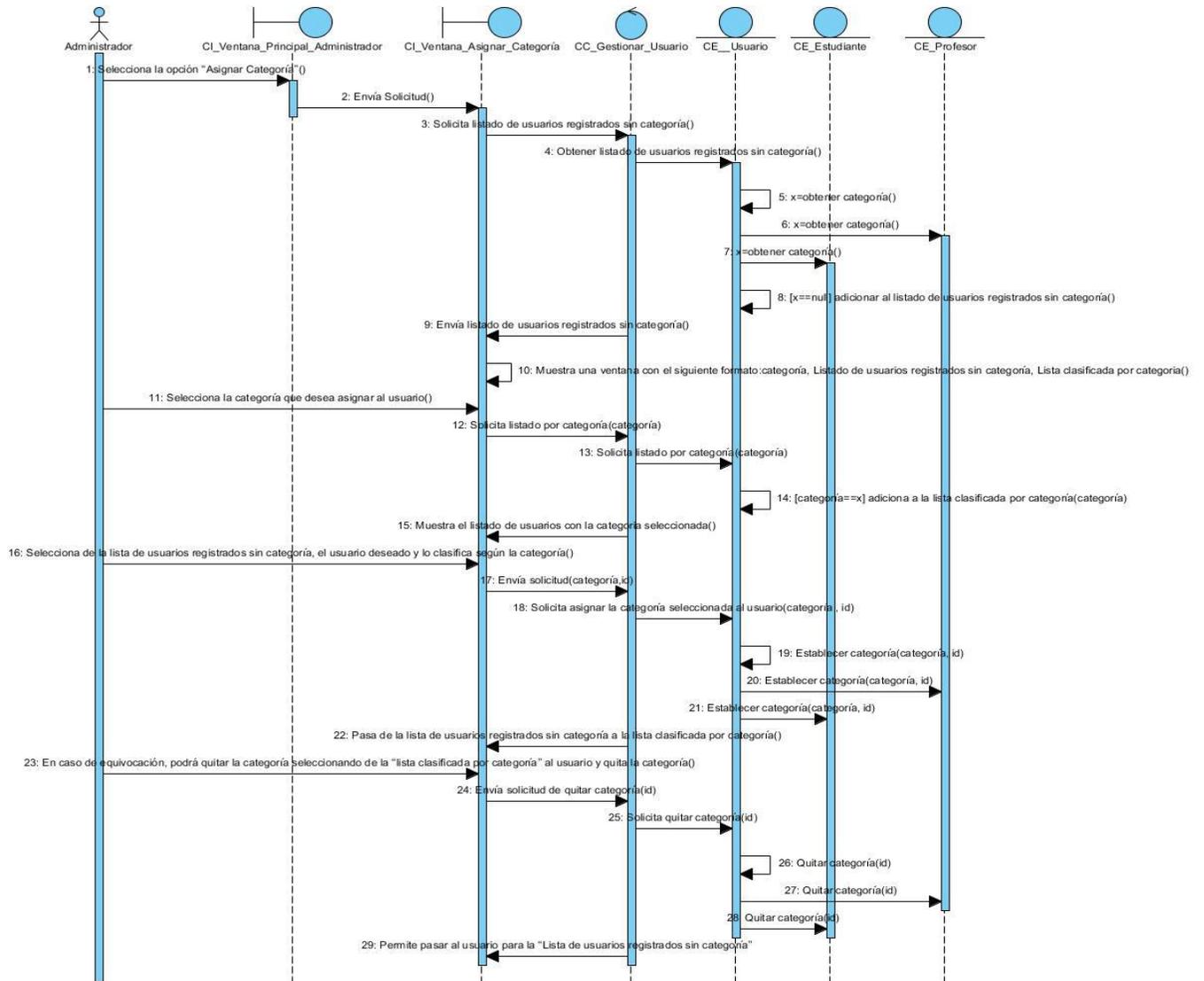


Figura 18. Diagrama de secuencia Asignar categoría.

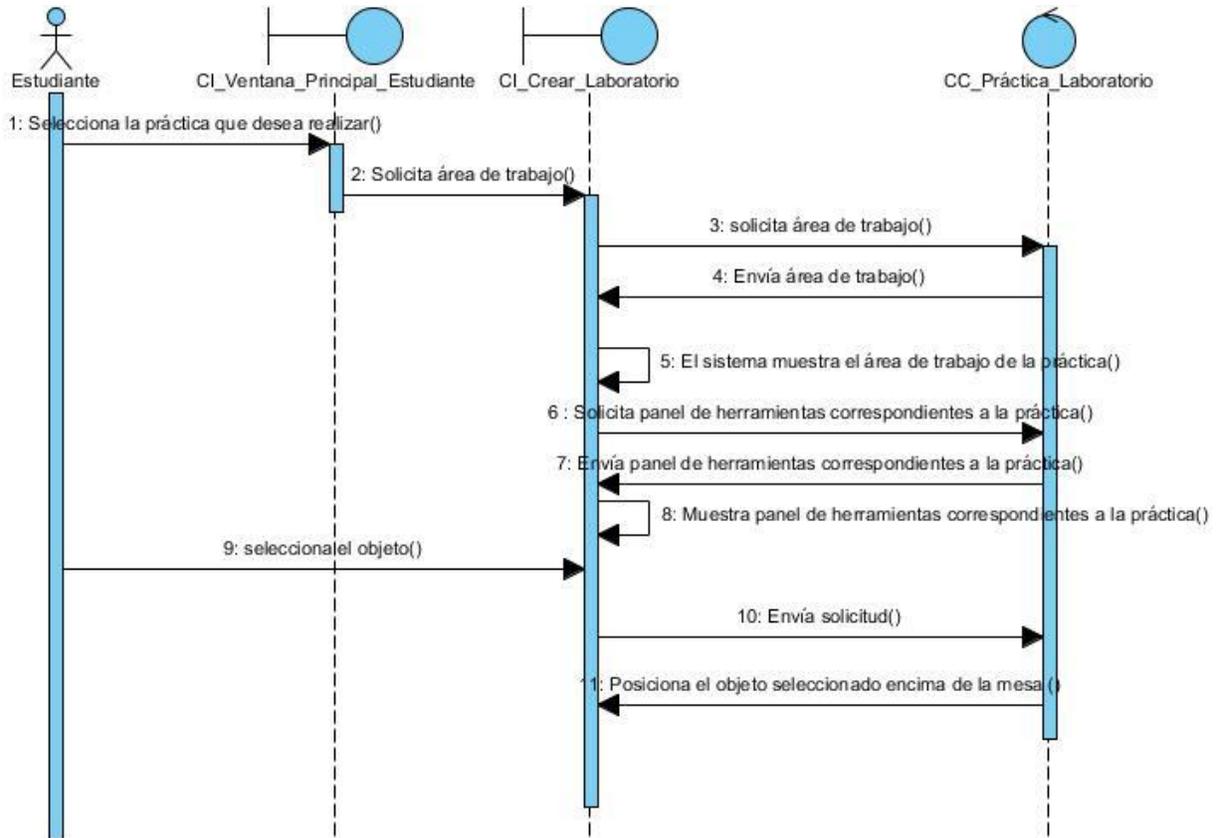


Figura 19. Diagrama de secuencia Crear laboratorio.

3.3.4 Diagrama de Despliegue

El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. (15)

Un Diagrama de Despliegue modela la arquitectura en tiempo de ejecución de un sistema. Esto muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se trazan en esos nodos. Los diagramas de despliegue representan los tipos de nodos del sistema y sus relaciones.

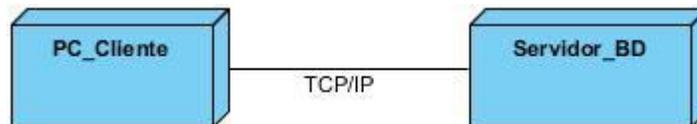


Figura 20. Diagrama de Despliegue.

3.3.5 Diagrama de clases persistentes

El Diagrama de Clases Persistentes presenta un mecanismo de implementación neutral para modelar los aspectos de almacenamiento de datos del sistema. Las clases persistentes, sus atributos, y sus relaciones pueden ser implementados directamente en una base de datos orientada a objetos. La persistencia es la capacidad de un objeto de mantener su valor en el espacio y en el tiempo. Es responsabilidad del diseñador definir cuáles clases son las que deben ser persistentes.

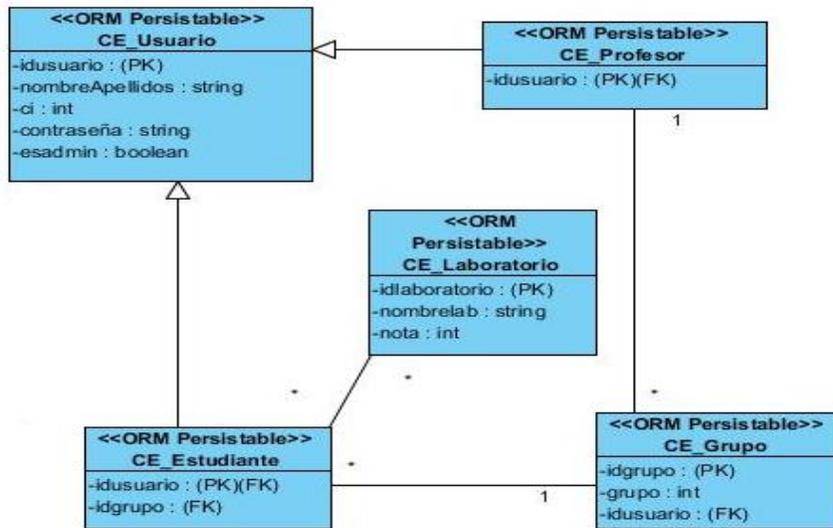


Figura 21. Diagrama de clases persistentes.

3.3.6 Modelo de Datos

El modelo de datos describe el comportamiento lógico y físico de los elementos persistentes de utilidad para dar soporte de información al sistema. Con el fin de garantizar la persistencia de los datos se confeccionó el modelo entidad relación, el cual se muestra a continuación.

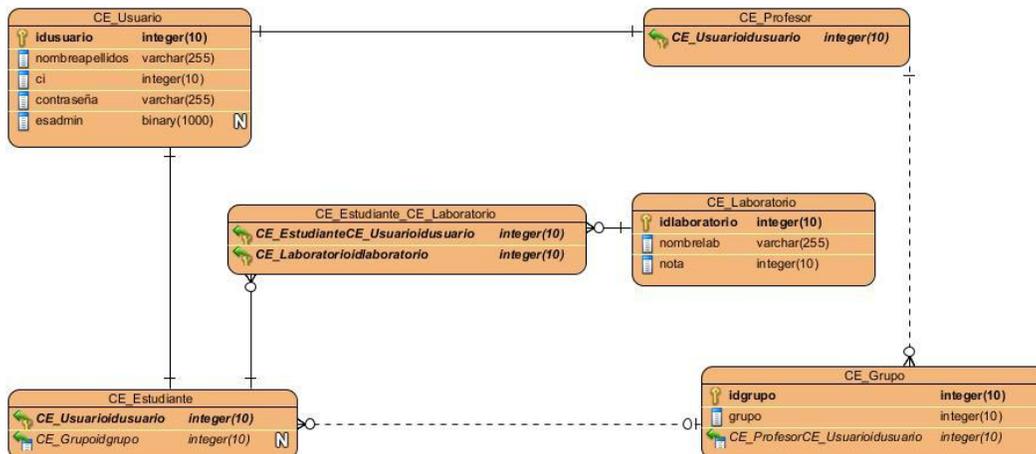


Figura 22. Modelo de datos.

3.4 Validación de la propuesta de diseño.

Una de las formas más habituales y convenientes de validar el sistema consiste en construir un prototipo, ya que es una versión operativa preliminar del sistema para fines de demostración y evaluación.

3.4.1 Propósitos del Prototipo

“En la fase de Análisis de un proyecto, su principal propósito es obtener y validar los requerimientos esenciales, manteniendo abiertas, las opciones de implementación. Esto implica que se debe tomar los comentarios de los usuarios, pero debemos regresar a sus objetivos para no perder la atención”. (30)

“En la fase de Diseño, su propósito, basándose en los requerimientos previamente obtenidos, es mostrar las ventanas, su navegación, interacción, controles y botones al usuario y obtener una retroalimentación que nos permite mejorar el Diseño de Interfaz”. (30)

3.4.2 Clases de Prototipo

Prototipo Corregido: Se refiere a una tabla experimental en donde el mismo sistema tiene la capacidad de auto corrección de errores, dentro de la misma. Prototipo Corregido

Prototipo No Funcional: Se utiliza principalmente para probar ciertos aspectos de diseño que no implican dentro de los objetivos del proyecto. Entrada, Procesamiento y Salida de información

Primer Prototipo de una serie: Sirve como una especie de modelo de prueba para la experimentación de los demás prototipos en relación con la interacción con el sistema.

Prototipo de características seleccionadas: Se desarrolla con el fin de dar a conocer un modelo con ciertas características parecidas a las características que tendrá el sistema final. Prototipo de características seleccionadas

3.4.3 Prototipo no funcional

Los prototipos de interfaz de usuario son una representación no funcional del sistema, que se le presenta al cliente para validar que cumple con todas las necesidades que el mismo requiere.

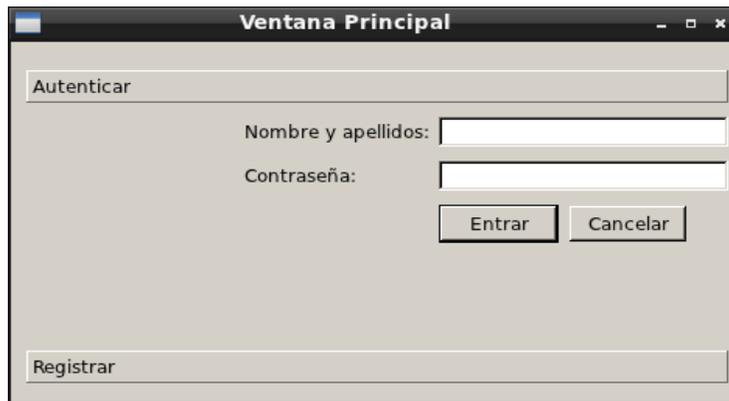
En la **Figura 23** se representa la interfaz de usuario que da respuesta al Caso de Uso del Sistema Registrar Usuario.



The screenshot shows a window titled "Ventana Principal" with a light gray background. At the top, there are two tabs: "Autenticar" (selected) and "Registrar". Below the tabs, there are three input fields: "Nombre y apellidos:", "Carnet ID:", and "Contraseña:". At the bottom right, there are two buttons: "Registrarse" and "Cancelar".

Figura 23. Interfaz Gráfica Registrar Usuario.

En la **Figura 24** se representa la interfaz de usuario que da respuesta al Caso de Uso del Sistema Autenticar Usuario.



The screenshot shows a window titled "Ventana Principal" with a light gray background. At the top, there are two tabs: "Autenticar" (selected) and "Registrar". Below the tabs, there are two input fields: "Nombre y apellidos:" and "Contraseña:". At the bottom right, there are two buttons: "Entrar" and "Cancelar".

Figura 24. Interfaz Gráfica Autenticar Usuario.

En la **Figura 25** se representa la interfaz de usuario Ventana Principal Administrador. Dónde están presente las opciones correspondientes: Asignar categoría y Eliminar usuario, las cuales permitirán hacer uso de estas funcionalidades

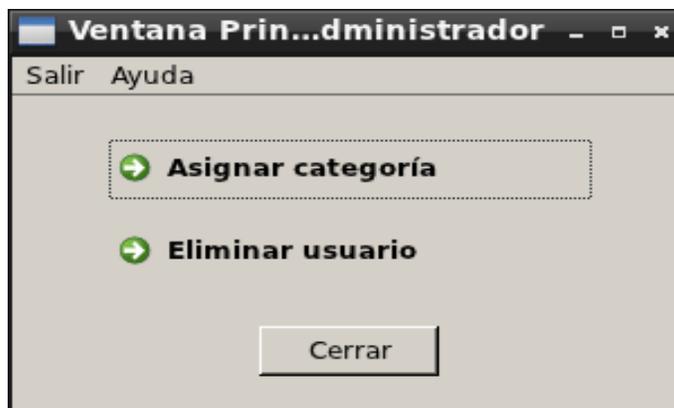


Figura 25. Interfaz Ventana Principal Administrador

En la **Figura 26** se representa la interfaz de usuario que da respuesta al Caso de Uso del Sistema Asignar categoría.

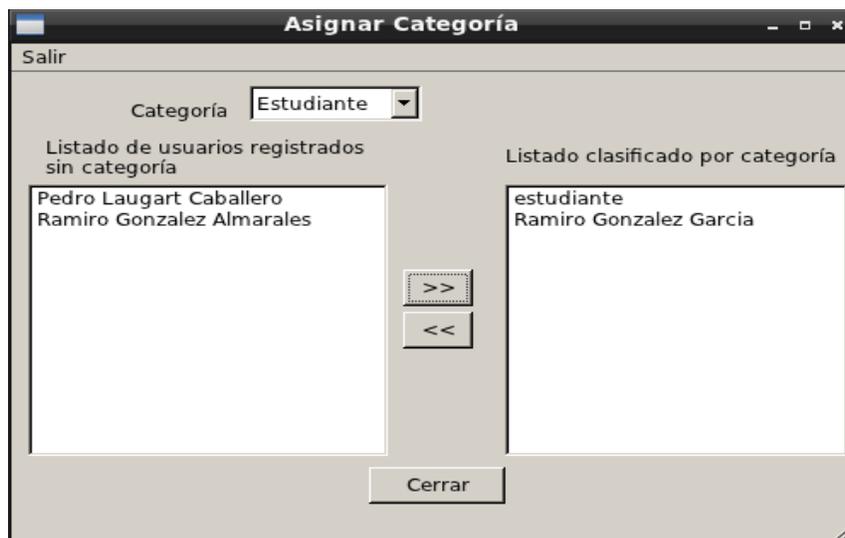


Figura 26. Interfaz Asignar categoría

En la **Figura 27** se representa la interfaz de usuario que da respuesta al Caso de Uso del Sistema Eliminar usuario.

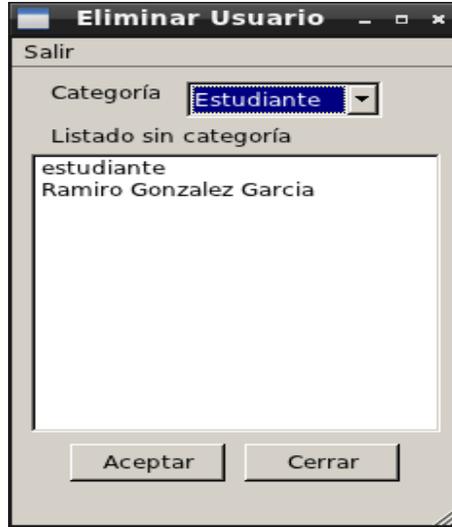


Figura 27. Interfaz Eliminar usuario

En la **Figura 28** se representa la interfaz de usuario que da respuesta al Caso de Uso del Sistema Crear laboratorio. Dónde está presente un formulario, en el cuál el usuario debe seleccionar la práctica que desea realizar, para poder acceder a la práctica en el sistema y tener acceso a Crear Laboratorio. Los temas son:

Las disoluciones acuosas de las sales. Concentración Másica

Separación de los componentes de las mezclas

Separación de los componentes de las mezclas. Vaporización y destilación.

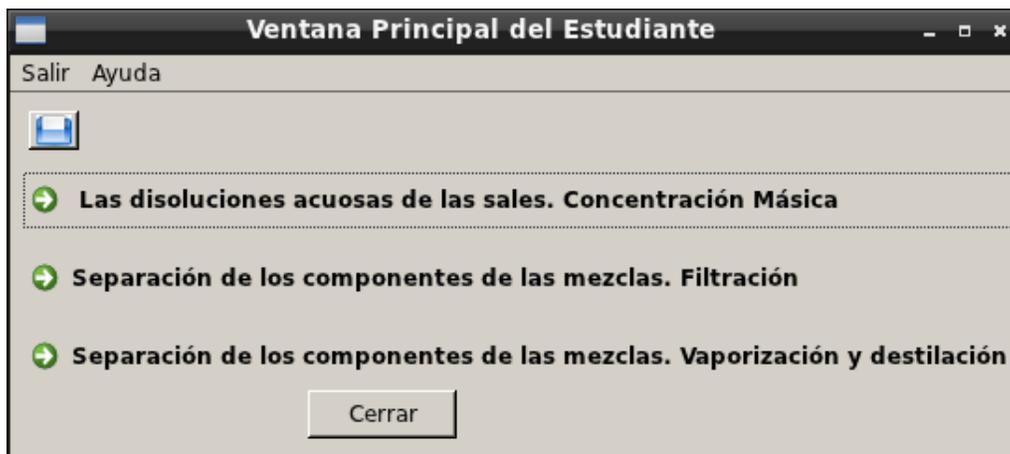


Figura 28. Interfaz Gráfica Ventana Principal Estudiante.

En la **Figura 29** se representa la interfaz de usuario que da respuesta al Caso de Uso del Sistema Crear laboratorio.



Figura 29. Interfaz Gráfica Crear Laboratorio

En la **Figura 30** se representa la interfaz de usuario Ventana Principal Profesor. Donde están presente las opciones correspondientes: Conformar grupo docente, Mostrar notas y Crear laboratorio, las cuales permitirán hacer uso de estas funcionalidades

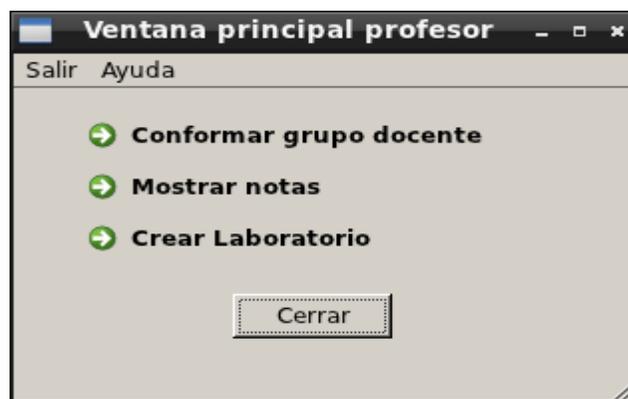


Figura 30. Interfaz Gráfica Ventana Principal Profesor

En la **Figura 31** se representa la interfaz de usuario que da respuesta al Caso de Uso del Sistema Conformar grupo docente.

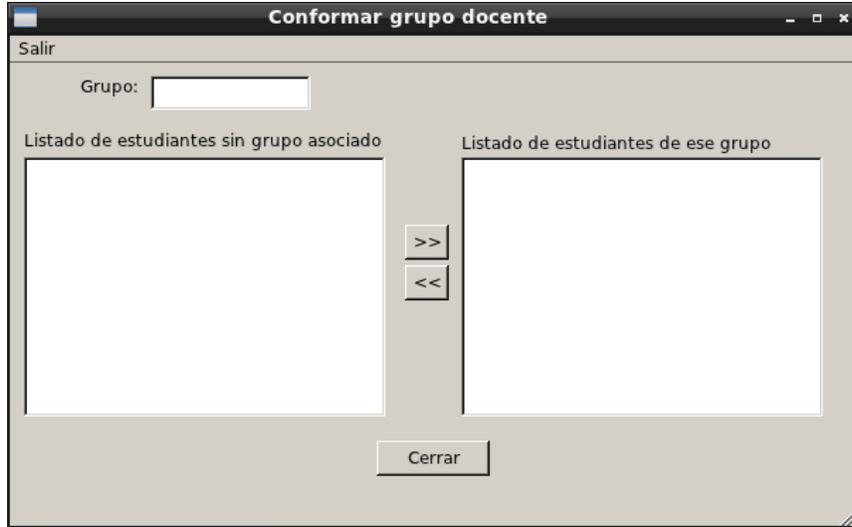


Figura 31. Interfaz Gráfica Conformar grupo docente

En la **Figura 32** se representa la interfaz de usuario que da respuesta al Caso de Uso del Sistema Mostrar notas.

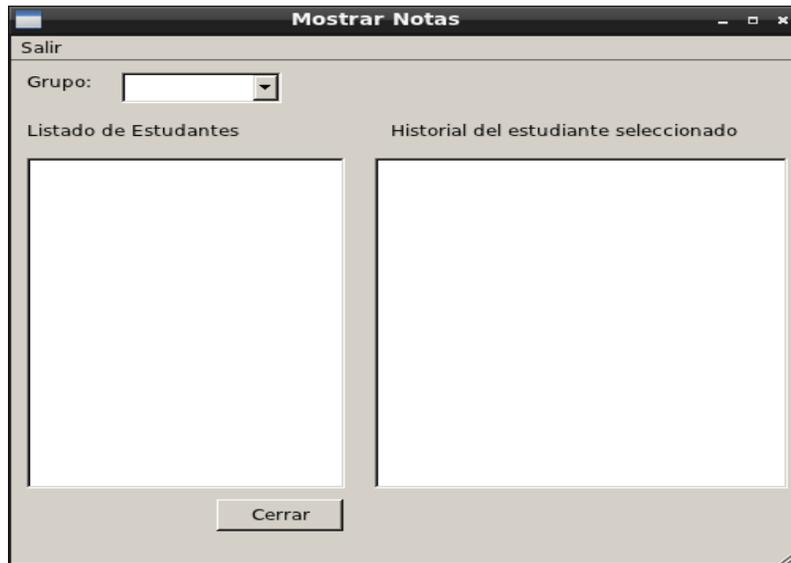


Figura 32. Interfaz Gráfica Mostrar notas.

Consideraciones Parciales

En este capítulo, se realizó una modelación del sistema en términos del análisis y diseño de los casos de uso. Como respuesta a la solución propuesta, se generaron los artefactos necesarios en el flujo de trabajo que propone RUP; identificándose las clases interfaces, de control y de entidad.

Capítulo 3: Análisis y Diseño del Sistema

Se obtuvo la realización de los casos de uso en el análisis y el diseño mostrando como resultado los diagramas de clases de análisis, diagramas de interacción por cada escenario de caso de uso, los diagramas de clases del diseño, despliegue, clases persistentes y modelo de datos.

Se aplicaron los patrones necesarios para lograr un diseño eficiente y se validó la propuesta de diseño mediante la elaboración un prototipo no funcional que permitió validar los requisitos y aspectos de la implementación.

Conclusiones generales

Como resultado de la investigación se arribaron a las siguientes conclusiones:

- Se realizó una investigación sobre los Laboratorios Virtuales existentes a nivel internacional, Cuba y la UCI, concluyendo que solo se pudo constatar una solución similar, pero no se pudo acceder al diseño de la misma.
- Se efectuó el estudio de las tecnologías y herramientas usadas en la actualidad para desarrollar aplicaciones de escritorio escogiendo las adecuadas para la realización de la propuesta.
- Se generaron los principales artefactos fundamentales propuestos por la metodología RUP, obteniendo la documentación necesaria para garantizar el futuro mantenimiento del sistema.
- Por último se realizó la validación de la propuesta a través de un demo concluyendo que la misma resuelve la situación problemática planteada.

Recomendaciones

Tomando como base la investigación realizada y la experiencia acumulada durante la realización de este trabajo, se proponen las siguientes recomendaciones:

1. Generalización e Implementación de la Práctica de Laboratorio Virtual de Química. En todos los centros de enseñanza media del país. Teniendo en cuenta la disponibilidad tecnológica y la flexibilidad de la propuesta.
2. Incluir nuevas funcionalidades al Laboratorio Virtual para la implementación de otras prácticas de la asignatura Química.
3. Utilizar el documento como material de consulta para investigaciones futuras.
4. Presentar el trabajo en eventos científicos con el objetivo de perfeccionarlo.

Bibliografía

1. **López García, Marta y Morcillo Ortega, Juan Gabriel.** *Las TIC en la enseñanza de la Biología en la educación secundaria: los laboratorios virtuales.* 2007.
2. **DORE, JASMIN y RAMOS, LUIS.** DEFINA Y EXPLIQUE EL CONCEPTO DE LABORATORIOS VIRTUALES CON TUTORIALES NORTEAMERICANOS. [En línea] jueves 15 de marzo de 2007. [Citado el: 10 de 12 de 2011.] <http://laboratoriovirtuallnorteamericano.blogspot.com/2007/03/defina-y-explique-el-concepto-de.html>.
3. **Mongue Najerá, Julián y Méndez Estrada, Víctor Hugo.** *Ventajas y Desventajas de usar laboratorios virtuales en educación a distancia.* s.l. : Revista Educación, 2007. ISSN:0379-7082.
4. **L. Rosado, J. R. Herreros.** *Nuevas aportaciones didácticas de los laboratorios virtuales y remotos en la enseñanza de la Física.* 2005. <http://www.uv.es/eees/archivo/286.pdf>.
5. **PATÓN, D.** Metodologías de Desarrollo de Software. [En línea] [Citado el: 15 de 06 de 2012.] <http://alarcos.inf-cr.uclm.es/doc/ISOFTWAREI/Tema04.pdf>.
6. **Canós, José H., Letelier, Patricio y Penad, M^a Carmen.** Metodologías Ágiles en el Desarrollo de Software. [En línea] [Citado el: 15 de 06 de 2012.] <http://www.willydev.net/descargas/prev/TodoAgil.pdf>.
7. **USMP.** RUP vs XP. [En línea] [Citado el: 15 de 06 de 2012.] <http://www.usmp.edu.pe/publicaciones/boletin/fia/info49/articulos/RUP%20vs.%20XP.pdf>.
8. **Ivar Jacobson, Graby Boosh, James Rumbaugh.** *El Proceso Unificado del Desarrollo de Software.*
9. **PALACIOS, Juan.** *El modelo Scrum.* . http://eva.uci.cu/file.php/161/Documentos/Materiales_complementarios/UD_1_Procesos/Metodologias/SCRUM/EI_modelo_SCRUM.pdf.
10. **Pressman, Roger S.** *Visual Paradigm. Ingeniería de Software, un enfoque práctico.* Quinta edición. s.l. : McGraw-Hill Companies, 2002. Vol. V, http://www.ecured.cu/index.php/Visual_Paradigm. ISBN: 8448132149..
11. **Jacobson, I., Booch, G. y Rumbaugh, J.** *El Proceso Unificado de Desarrollo de Software.* s.l. : Stanford: Addison – Wesley Company, 2000.
12. **Juan Carlos Guevara Bolaños, Lely Luengas Contreras.** *Laboratorio virtual para la destilación química.* Universidad Distrital Colombia. 2009. http://www.tise.cl2009tise_2009pdf9.pdf.
13. **Rodríguez., Mirurgia Avila.** *Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas.*

14. **Pressman, Roger S.** *Ingeniería de Software, un enfoque práctico*. Quinta edición. s.l. : McGraw-Hill Companies, 2002. ISBN: 8448132149.
15. **González, Victorino Blanco.** La instalación y configuración de Eclipse, un entorno de desarrollo integrado (IDE) que sirve para muchos lenguajes. Eclipse tiene interesantes utilidades para Java y además es gratuito. [En línea] 2004. [Citado el: 10 de 12 de 2011.] <http://www.desarrolloweb.com/articulos/1692.php>.
16. **González Barahona, Jesús M.** Introducción al software libre. Capítulo 9. Estudio de casos. [En línea] [Citado el: 10 de 12 de 2011.] http://www.atenas.cult.cu/ri/informatica/manuales/sl/introduccion_al_SL/eclipse.html.
17. **PaBLoX, Por.** Qt Creator, desarrollando aplicaciones rápidamente. [En línea] 2009. [Citado el: 15 de 01 de 2012.] <http://www.glatelier.org/2009/05/qt-creator-desarrollando-aplicaciones-rapidamente/>.
18. **Developer Network.** SITIO OFICIAL QTCREATOR. [En línea] 2011. [Citado el: 15 de 01 de 2012.] http://developer.qt.nokia.com/wiki/Category:Tools::QtCreator_Spanish.
19. **Autodesk.** Productos Autodesk 3ds Max. [En línea] [Citado el: 20 de 01 de 2012.] <http://www.autodesk.es/adsk/servlet/pc/index?siteID=455755&id=14626995>.
20. **3D INFOGRÁFICA.** 3D INFOGRÁFICA. [En línea] 2011. [Citado el: 22 de 01 de 2012.] <http://www.3dinfografica.com/v2/destacados/noticia.php?nid=199>.
21. **Moya, Rafael.** Guía para Modelar arquitectura en Blender. [En línea] 2010. [Citado el: 22 de 01 de 2012.] <https://docs.google.com/viewer?a=v&pid=explorer&chrome=true&srcid=0BwC9xl05D8nBMWQwOTVjYjctMjBhMi00MTM0LTljZGQtNTQ1MTk1YWQ2OTM5&hl=en>.
22. **webestilo.** Introducción a MySQL. [En línea] [Citado el: 25 de 01 de 2012.] <http://www.webestilo.com/mysql/intro.phtml>.
23. **aulafacil.** MySQL. [En línea] [Citado el: 28 de 01 de 2012.] <http://www.aulafacil.com/mysql/curso/Lecc-1.htm>.
24. **DENZER, PATRICIO.** PostgreSQL : s.n., 23 de octubre de 2002. <http://profesores.elo.utfsm.cl/~agv/elo330/2s02/projects/denzer/informe.pdf>..
25. **Maldonado, Daniel Martin.** SQLite, el motor de base de datos ágil y robusto. [En línea] 1 de julio de 2008. [Citado el: 28 de 01 de 2012.] <http://www.aplicacionesempresariales.com/sqlite-el-motor-de-base-de-datos-agil-y-robusto.html>.
26. **Jonás A. Montilva C., Ph.D.** *Modelo del Negocio*. Mérida – Venezuela : s.n., 2007. <http://kuainasi.ciens.ucv.ve/ideas07/documentos/conferencias/ConferenciaJonasMontilva.pdf>.
27. **Larman, Craig.** UML y Patrones. . La Habana : s.n., 2004.

28. **Pressman, Roger S.** Ingeniería Del Software: Un Enfoque Practico. *Ingeniería Del Software: Un Enfoque Practico*. 5ta Edición. 2001.
29. **Jacobson, Ivar, Booch, Grady and Rumbaugh, James.** El Proceso Unificado de Desarrollo. [ed.] Addison Wesley. *El Proceso Unificado de Desarrollo*. 2002.
30. **E. KENDALL KENNETH, E. KENDALL JULIE, ANTONIO NÚÑEZ.** Análisis y diseño de sistemas. 6ta edición. s.l. : Pearson Educación, 2005, pág. 726 páginas.
31. **Ciberhábitat, Laboratorios virtuales en la educación. Introducción -.**
<http://www.ciberhabitat.gob.mx/universidad/ui/esyti/lv1.htm>.
32. **Morales., Rosana Camacho.** *Ingeniería Inversa del laboratorio virtual Ensamblaje de un Computador*. s.l. : Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas.
33. **Basset, Dunia Navarro.** *PROPUESTA DE DISEÑO DE UN LABORATORIO VIRTUAL DE FÍSICA PARA ESTUDIANTES DE INGENIERÍA EN CIENCIAS INFORMÁTICAS*. s.l. : Trabajo de Diploma para optar por el título de Ingeniero Informático., 2011.
34. **Meléndez, Wilder López.** Datos en SQL Server 2000.x. [En línea] [Citado el: 1 de 02 de 2012.] <http://www.geocities.ws/wlopezm/articulos/migrar.pdf>.
35. **Suárez, Ascánder.** DESARROLLO ÁGIL DE SOFTWARE – CASO PROGRAMACIÓN EXTREMA - XP. Ingeniería de Software II. [En línea] Trimestre Abril/Julio de 2009. [Citado el: 3 de 02 de 2012.] <http://ldc.usb.ve/~abianc/materias/ci4713/metodologiasagiles.pdf>.
36. **Aportes.** Simulación: el laboratorio virtual : Química. [En línea] [Citado el: 5 de 02 de 2012.] http://aportes.educ.ar/quimica/nucleo-teorico/tradiciones-de-ensenanza/como-ensenamos/simulacion_el_laboratorio_virt.php?page=2.
37. **disenodesistemasjrc.** Herramienta CASE. [En línea] [Citado el: 5 de 02 de 2012.] <https://sites.google.com/site/disenodesistemasjrc/herramientas-case-y-uml>.
38. **merinde.** modelo de casos de uso del sistema. [En línea] [Citado el: 10 de 02 de 2012.] http://merinde.net/index.php?option=com_content&task=view&id=89.
39. **JACOBSON, Ivar, BOOCH, Grady y RUMBAUGH, James.** *El Proceso Unificado de Desarrollo de Software*. Pearson Education. Madrid : s.n., 2000. ISBN 84-7829-036-2..
40. Lenguaje de Modelaje Unificado (UML). [En línea] [Citado el: 1 de 02 de 2012.] http://www.ecured.cu/index.php/Lenguaje_de_Modelaje_Unificado.
41. **buenastareas.** Scrum: Metodología Agil De Desarrollo De Software. [En línea] [Citado el: 10 de 02 de 2012.] <http://www.buenastareas.com/ensayos/Scrum-Metodologia-Agil-De-Desarrollo-De/2655970.html>.

42. —. Scrum: Metodología Agil De Desarrollo De Software. [En línea] [Citado el: 10 de 02 de 2012.] <http://www.buenastareas.com/ensayos/Scrum-Metodologia-Agil-De-Desarrollo-De/2655970.html>.
43. —. Diagramacion Uml. [En línea] [Citado el: 10 de 02 de 2012.] <http://www.buenastareas.com/ensayos/Diagramacion-Uml/242411.html>.
44. **etnassoft**. ScrumManager: Gestión de proyectos. [En línea] [Citado el: 15 de 02 de 2012.] <http://www.etnassoft.com/biblioteca/scrummanager-gestion-de-proyectos/>.
45. —. ScrumManager: Gestión de proyectos. [En línea] [Citado el: 15 de 02 de 2012.] <http://www.etnassoft.com/biblioteca/scrummanager-gestion-de-proyectos/>.
46. **Pérez[MSP], Fabián Bermeo**. Metodología RUP - desarrollo de software de calidad. [En línea] 8 de diciembre de 2010. [Citado el: 15 de 02 de 2012.] <http://fabianbermeop.blogspot.com/2010/12/metodologia-rup-desarrollo-de-software.html>.
47. —. Metodología RUP - desarrollo de software de calidad. [En línea] miércoles 8 de diciembre de 2010. [Citado el: 15 de 02 de 2012.] <http://fabianbermeop.blogspot.com/2010/12/metodologia-rup-desarrollo-de-software.html>.
48. **Ing. Marcelo Rea, Silvia Odóñez, Marco Navas, Fernando Martínez**. Características de la Metodología RUP. [En línea] [Citado el: 17 de 02 de 2012.] http://esenciaypotencia.com/index.php?option=com_content&view=article&id=67&Itemid=75.
49. —. Características de la Metodología RUP. [En línea] [Citado el: 17 de 02 de 2012.] http://esenciaypotencia.com/index.php?option=com_content&view=article&id=67&Itemid=75.
50. **tablondeanuncios**. Blender: Herramienta De Modelado Y Animación 3D. [En línea] [Citado el: 18 de 02 de 2012.] http://www.tablondeanuncios.com/libre_expresion/historia_ver.php/blender_herramienta_de_modelado_y_animacion_3d-1113.
51. **Gracia, Joaquin**. Patrones de diseño. [En línea] 27 de Mayo de 2005. [Citado el: 23 de 4 de 2012.] <http://www.ingenierosoftware.com/analisisydiseno/patrones-diseno.php>.

Anexos

Diagramas de Clases del Análisis

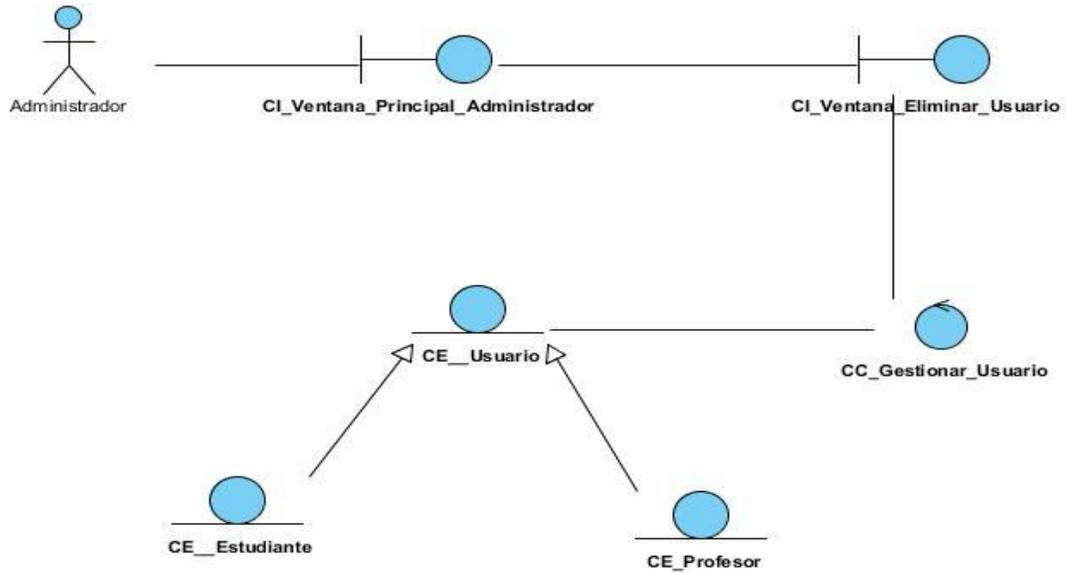


Figura 33. Diagrama de clases del análisis Eliminar usuario.

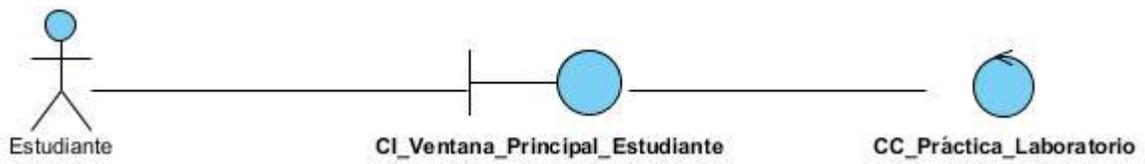


Figura 34. Diagrama de clases del análisis Mostrar escena.

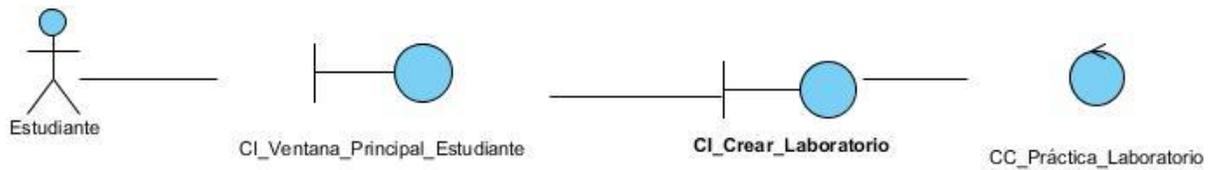


Figura 35. Diagrama de clases del análisis Administrar objeto.

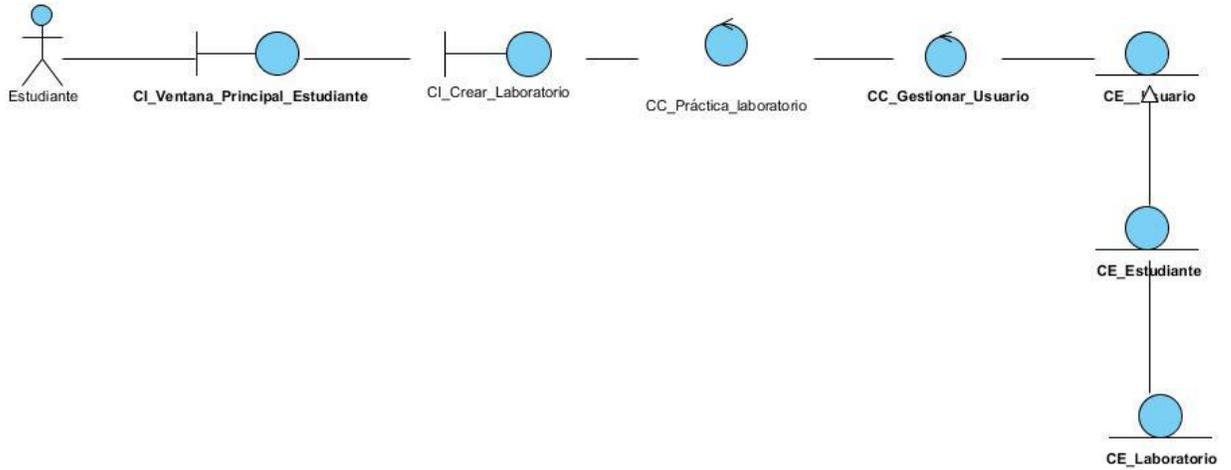


Figura 36. Diagrama de clases del análisis Devolver resultado de la práctica.

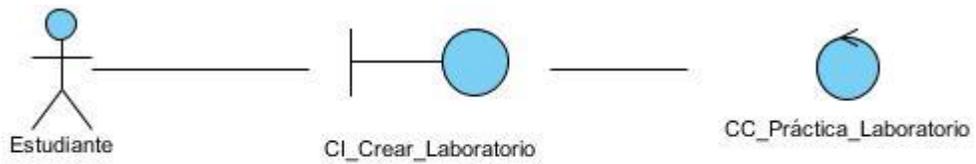


Figura 37. Diagrama de clases del análisis Guardar proyecto.

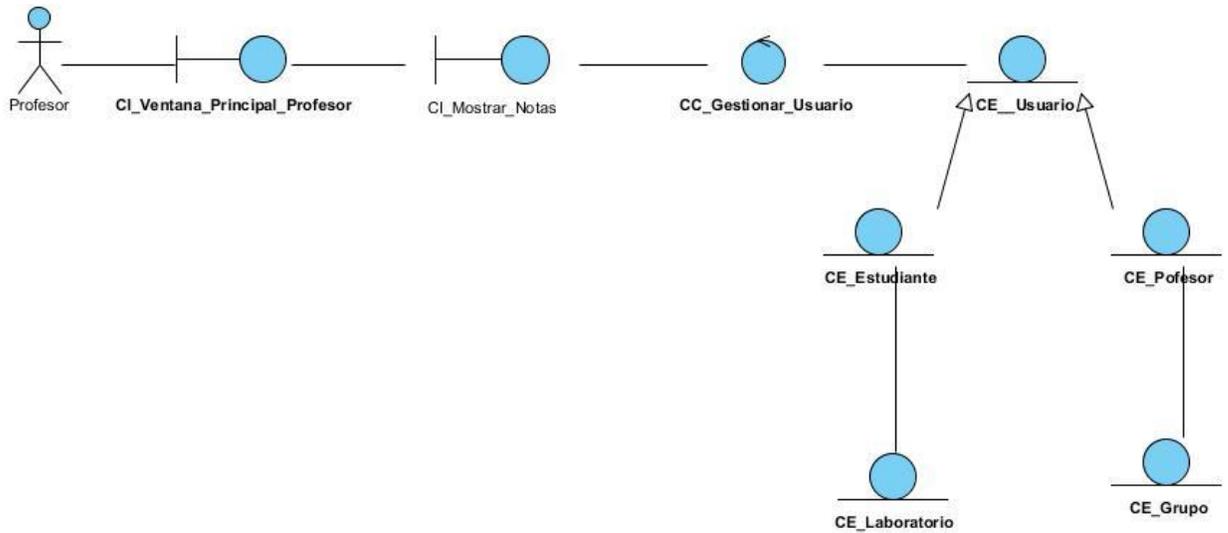


Figura 38. Diagrama de clases del análisis Mostrar notas.

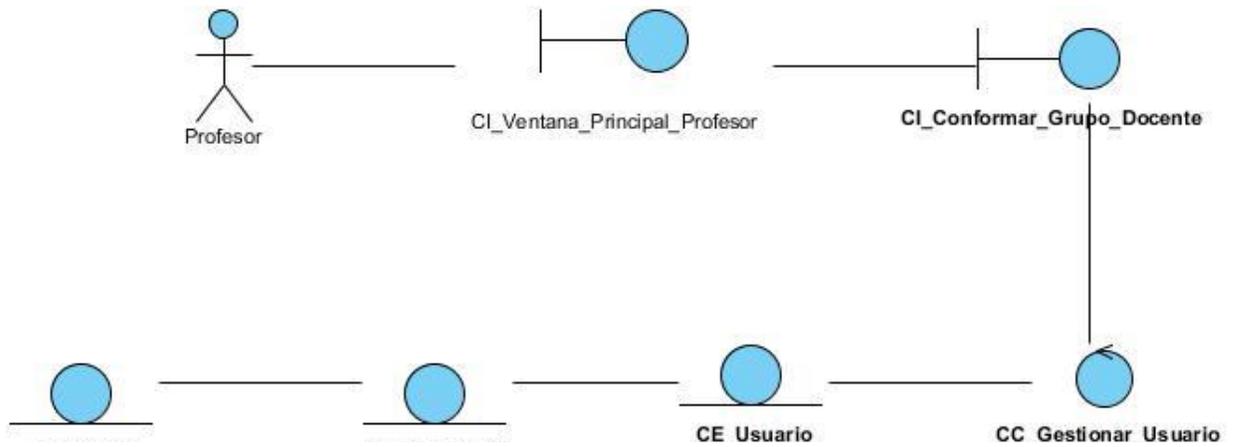


Figura 39. Diagrama de clases del análisis Conformar grupo docente.

Diagramas de Colaboración

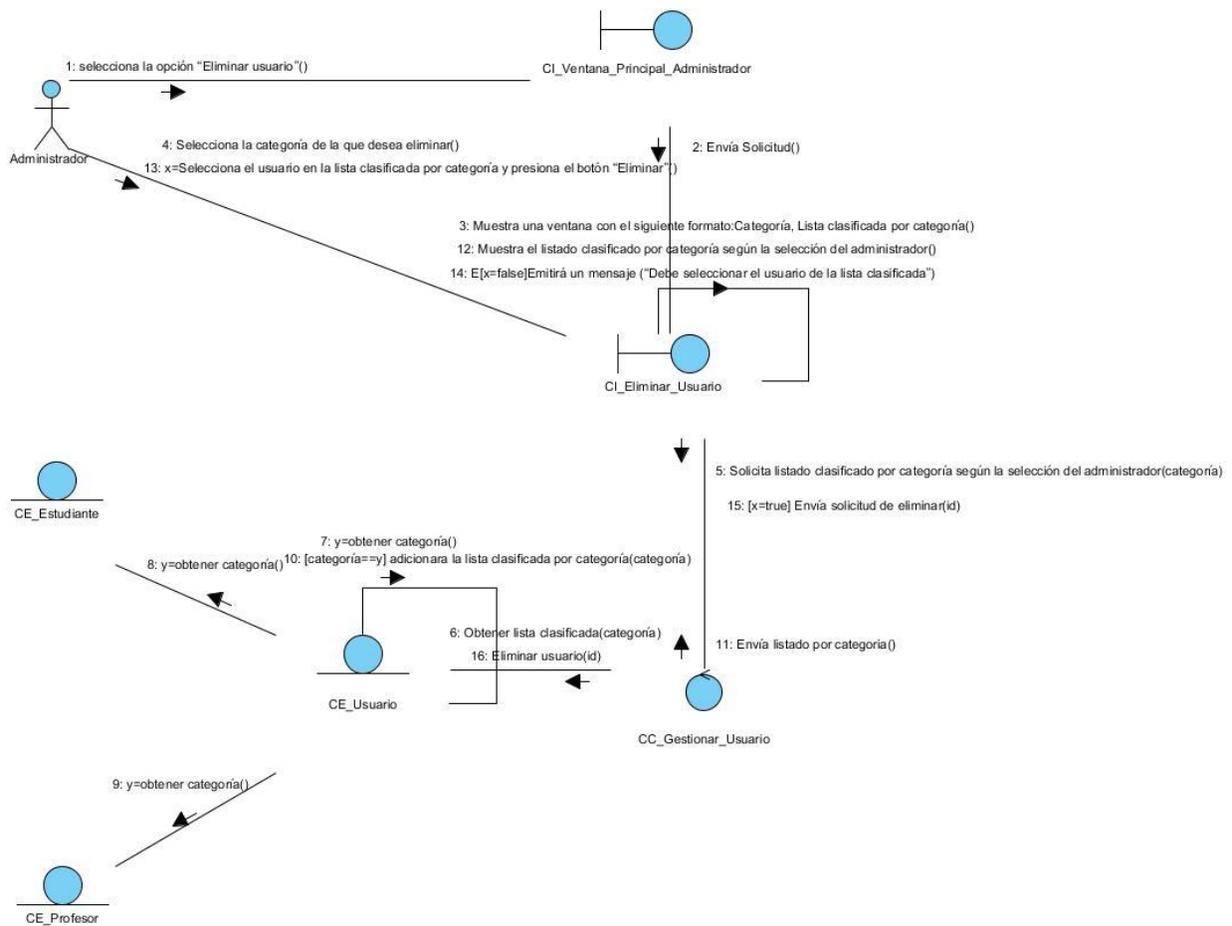


Figura 40. Diagrama de colaboración Eliminar usuario.



Figura 41. Diagrama de colaboración Mostrar escena.

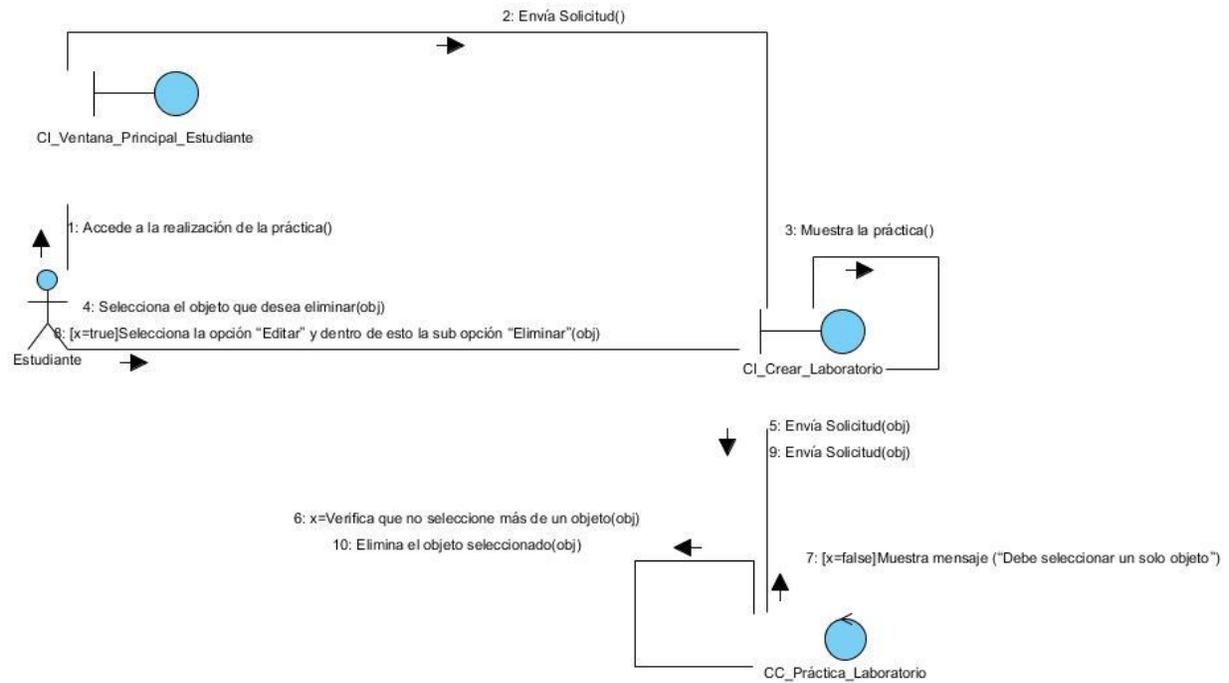


Figura 42. Diagrama de colaboración Administrar objeto, sección Eliminar objeto.

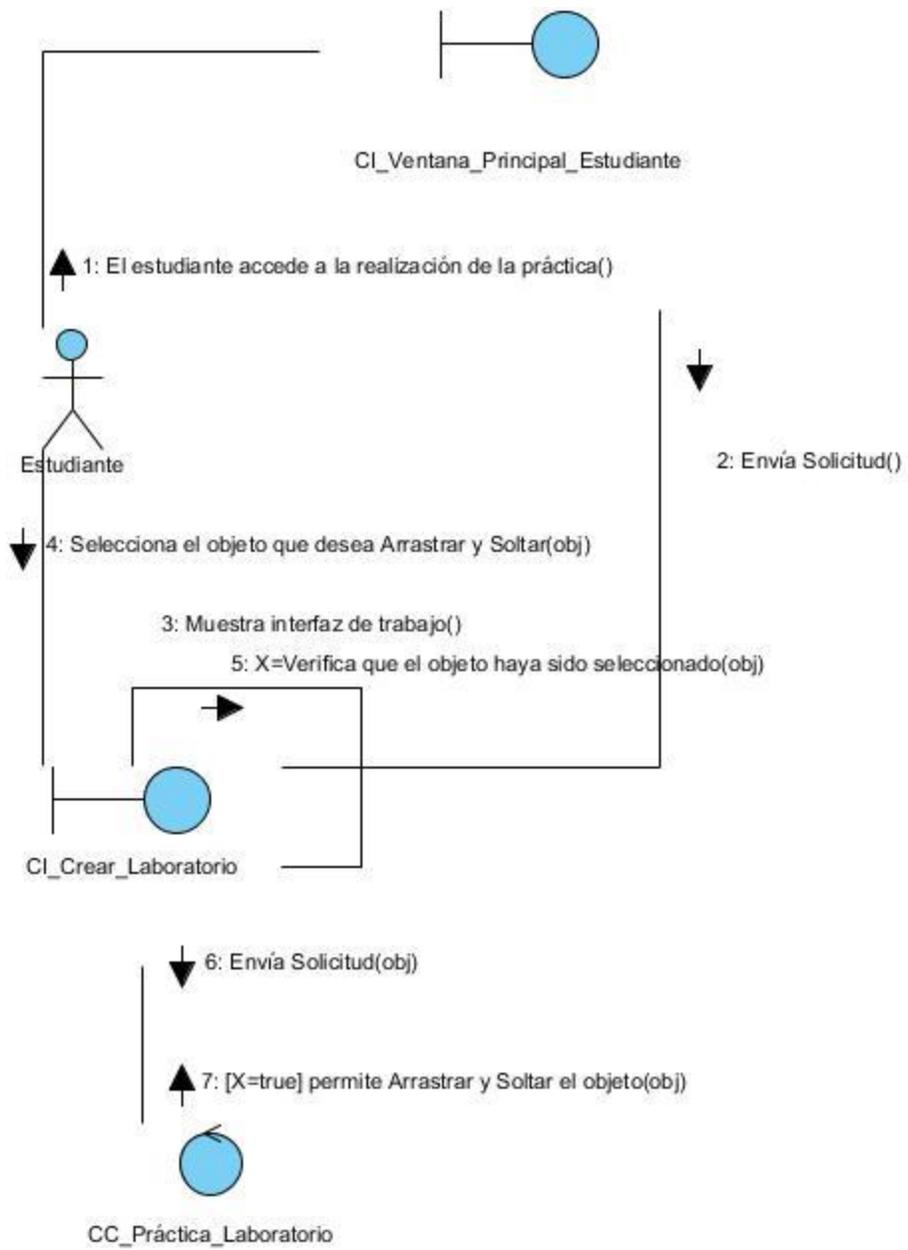


Figura 43. Diagrama de colaboración Administrar objeto, sección Arrastrar y soltar objeto.

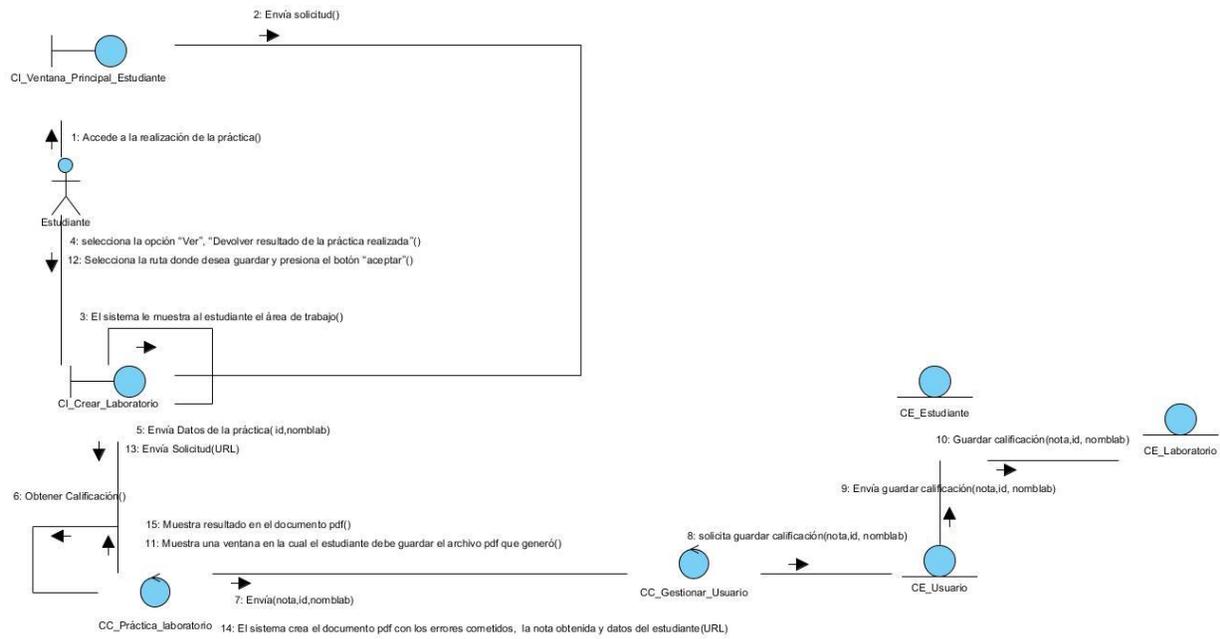


Figura 44. Diagrama de colaboración Devolver resultado de la práctica realizada.

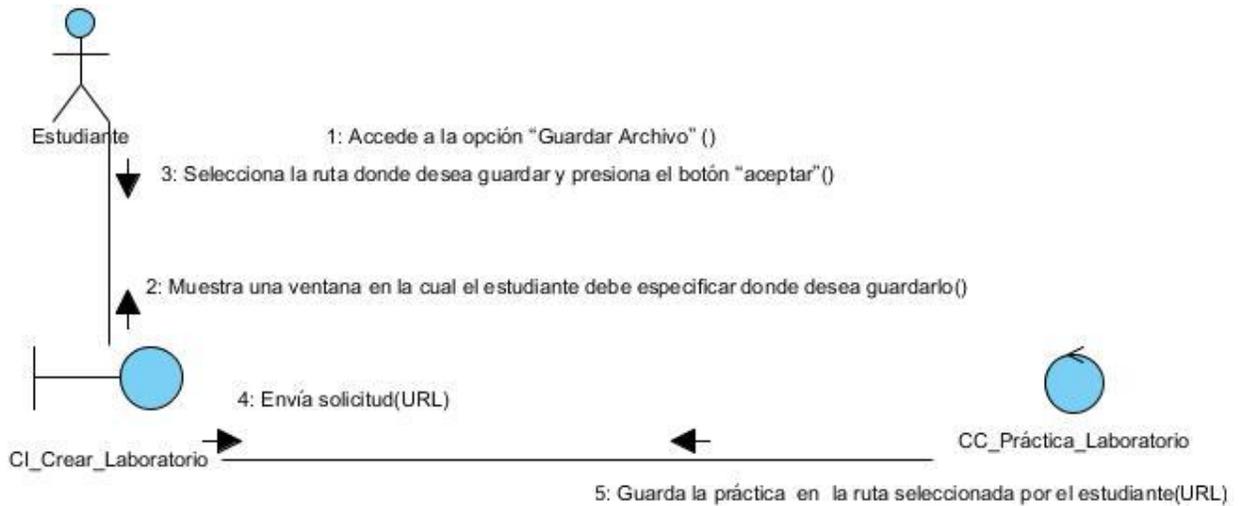


Figura 45. Diagrama de colaboración Guardar proyecto.

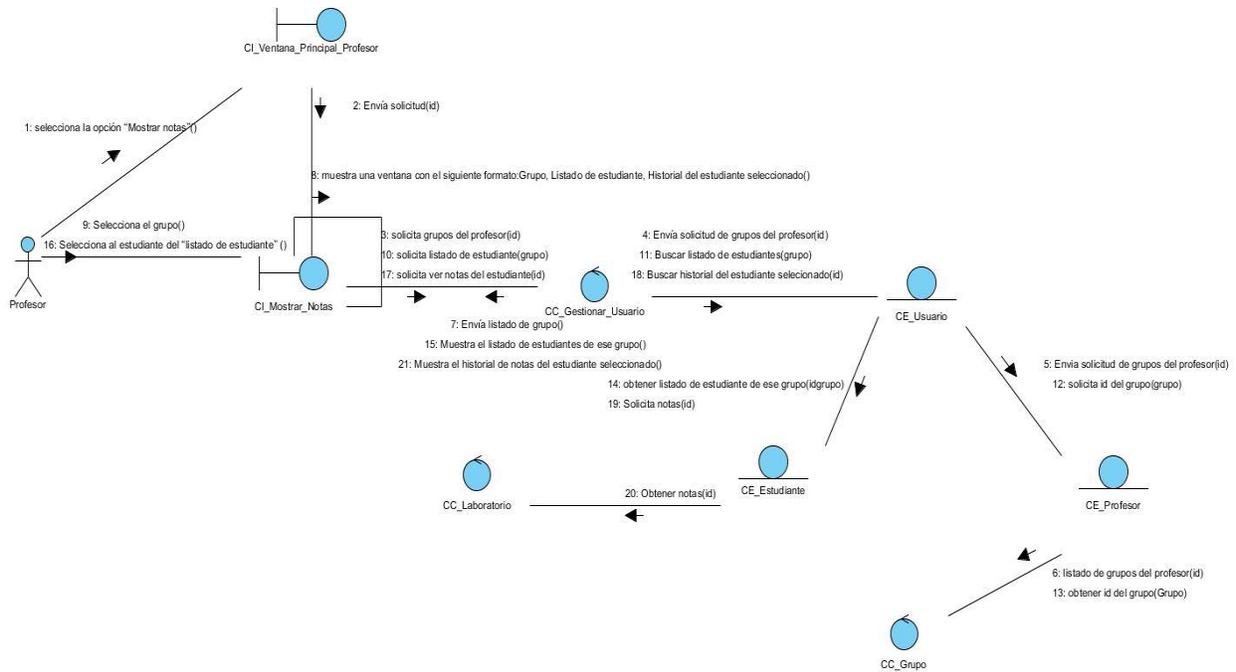


Figura 46. Diagrama de clases del diseño Mostrar notas.

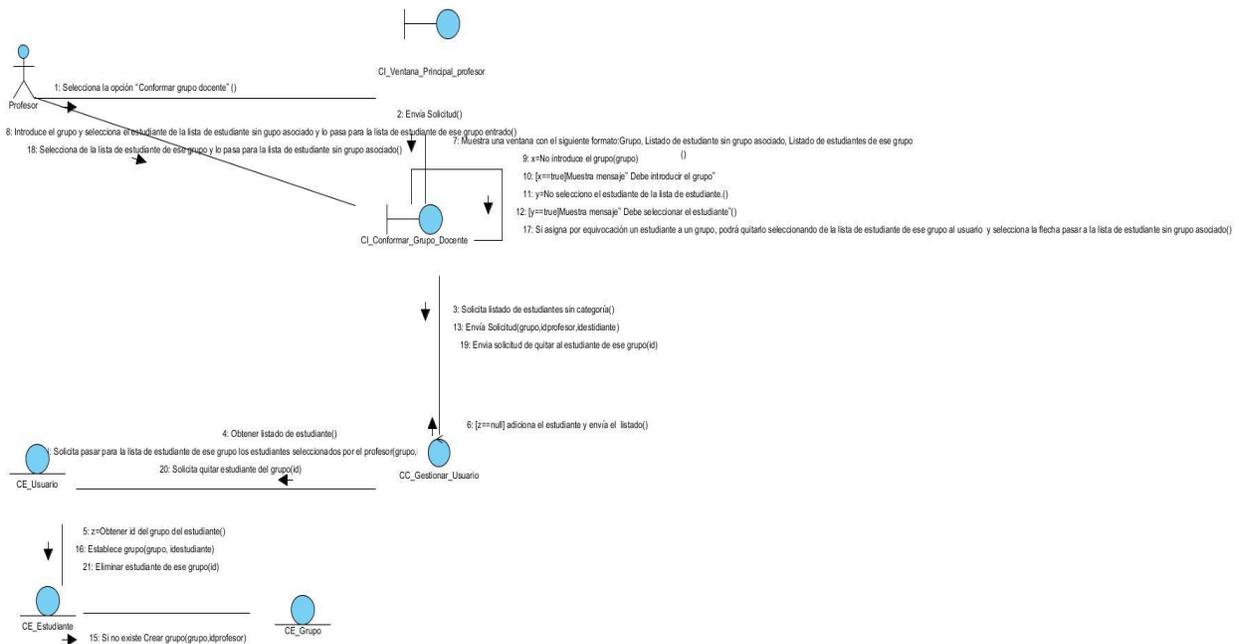


Figura 47. Diagrama de colaboración Conformar grupo docente.

Diagramas de Secuencias del Diseño

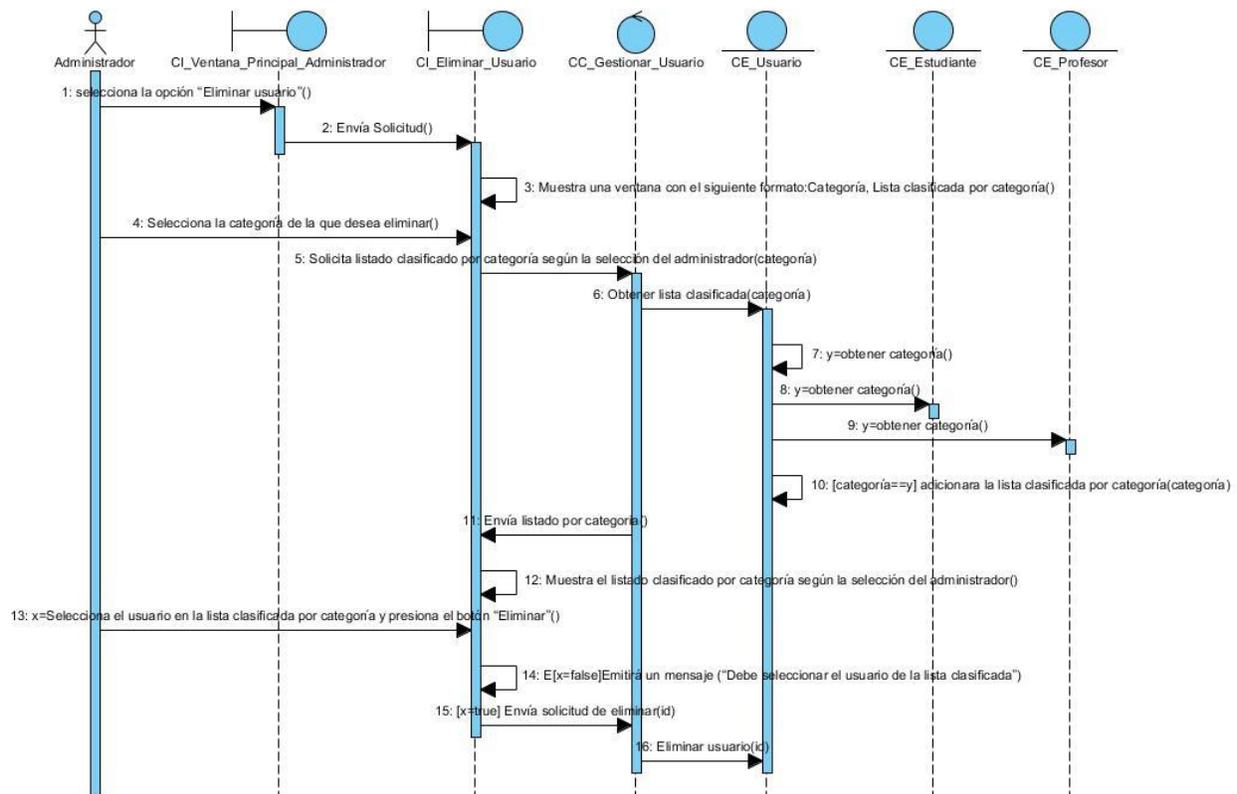


Figura 48. Diagrama de secuencia Eliminar usuario.

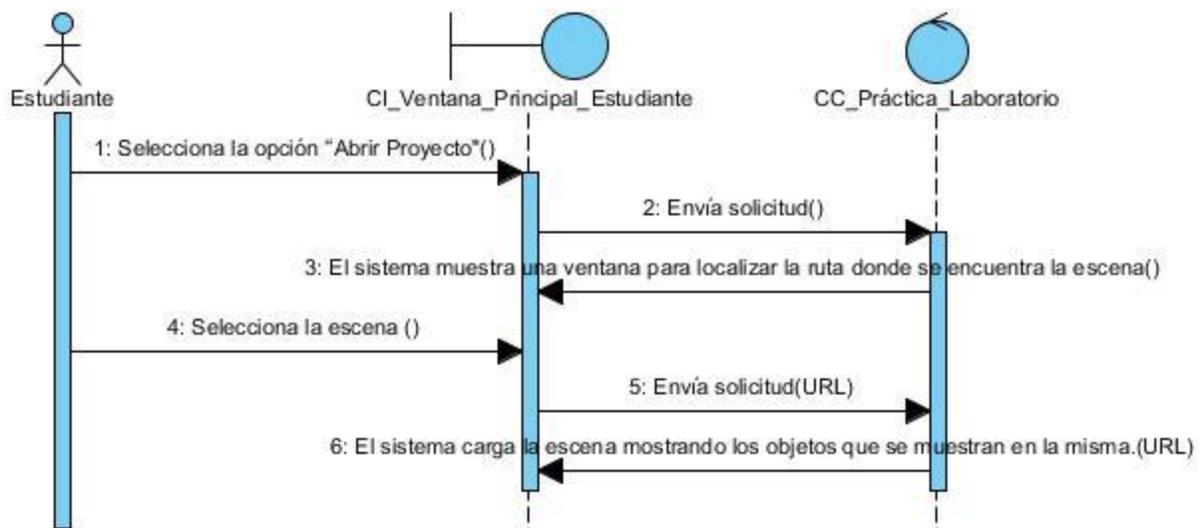


Figura 49. Diagrama de secuencia Mostrar la escena guardada.

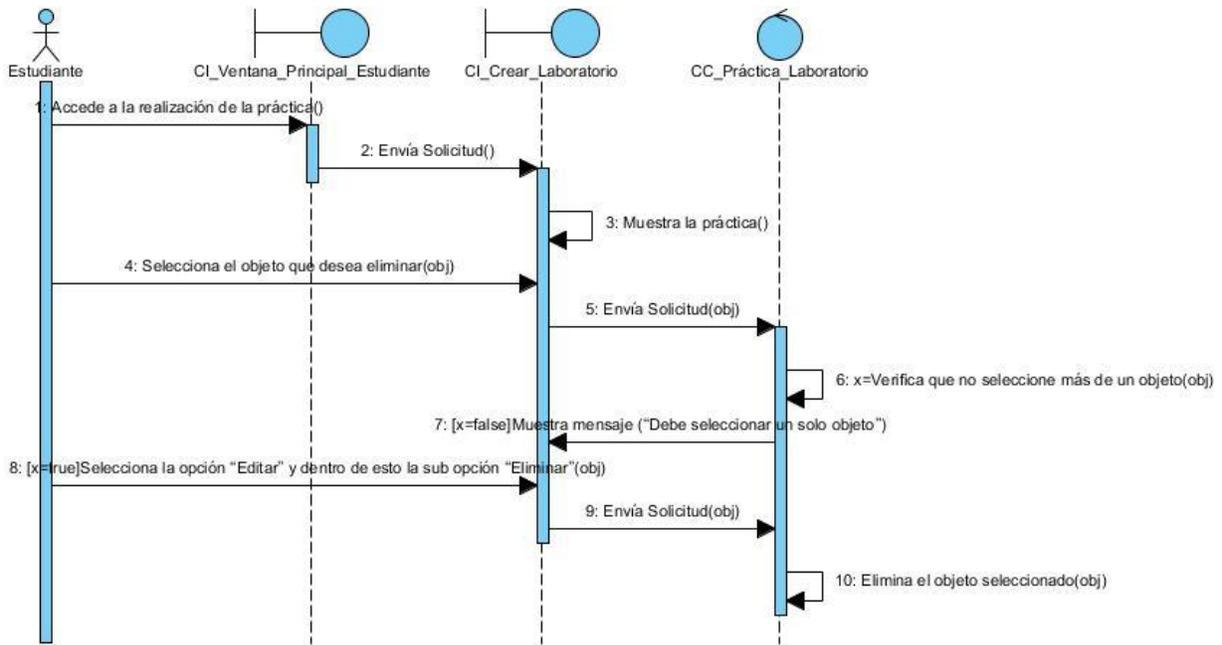


Figura 50. Diagrama de secuencia Administrar objeto, Sección "Eliminar objeto".

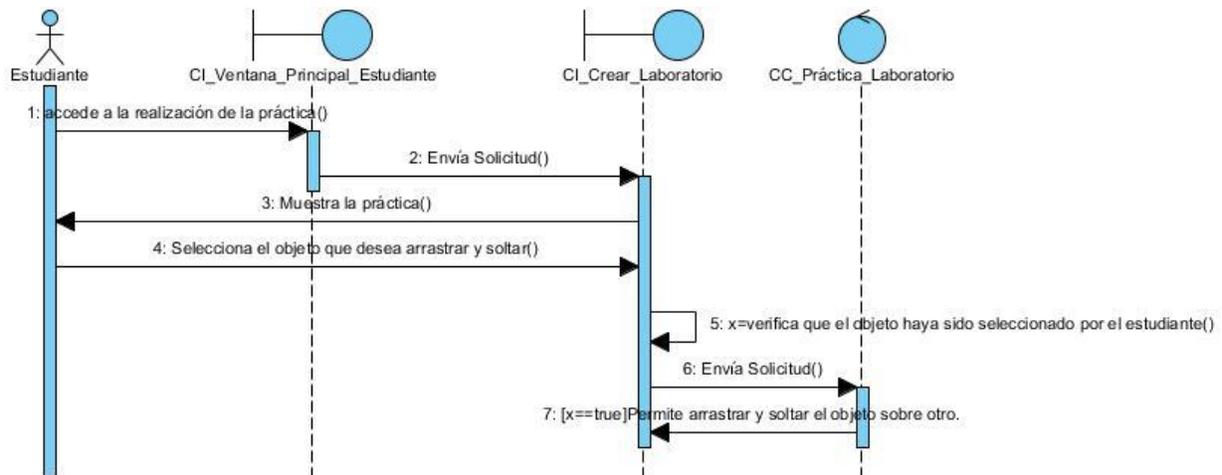


Figura 51. Diagrama de secuencia Administrar objeto, Sección "Arrastrar y soltar objeto".

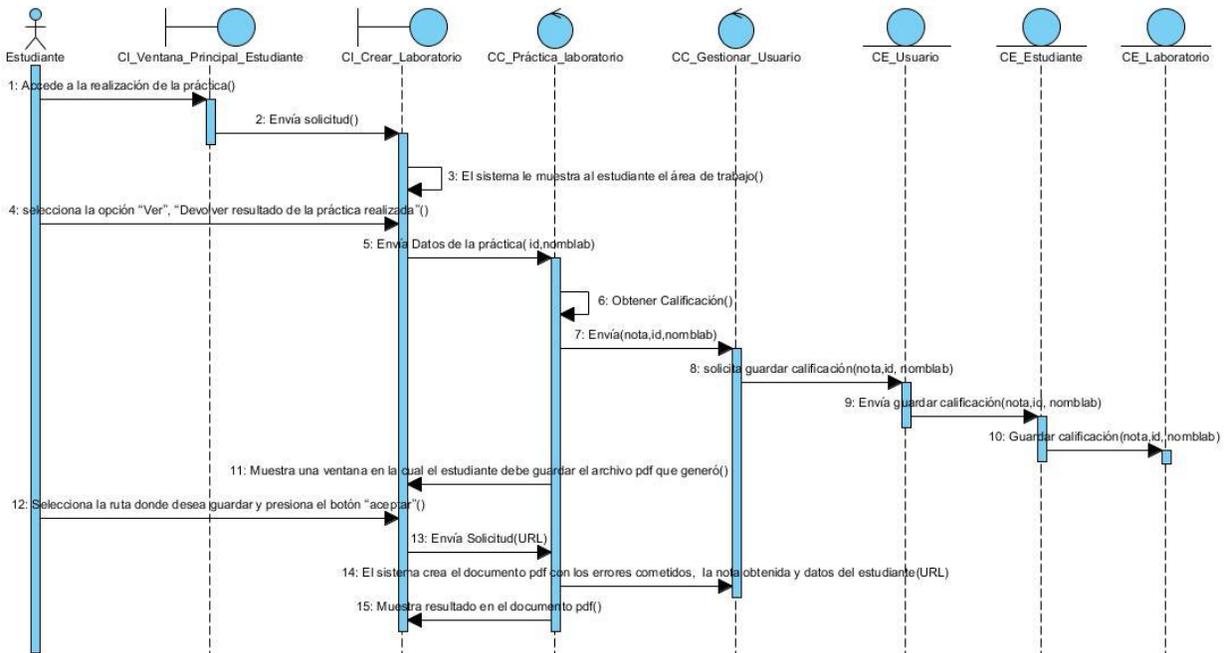


Figura 52. Diagrama de secuencia Devolver resultado de la práctica realizada.

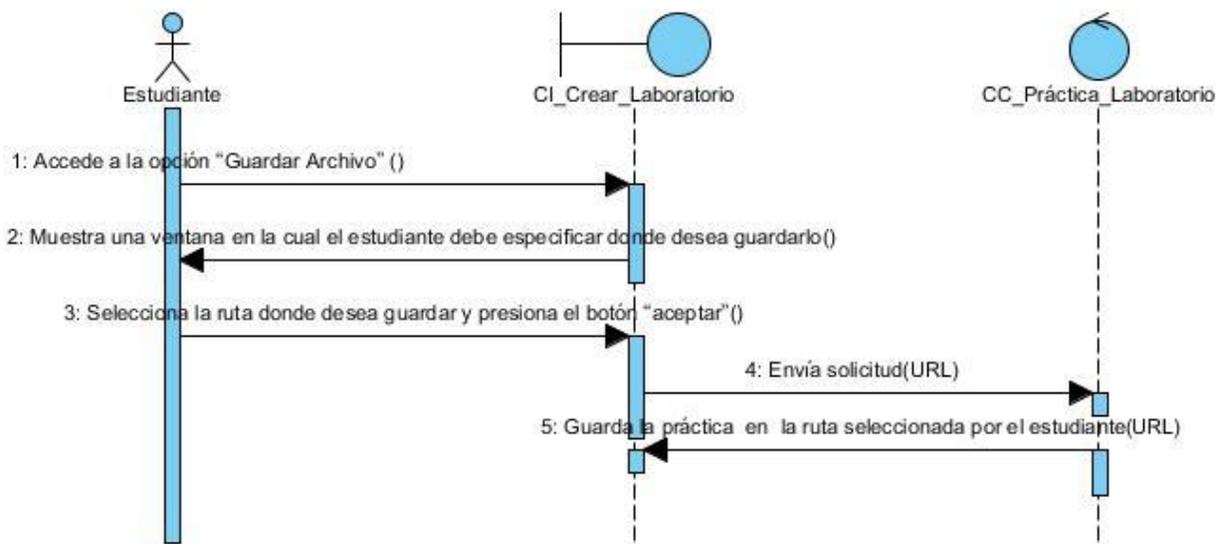


Figura 53. Diagrama de secuencia Guardar proyecto.

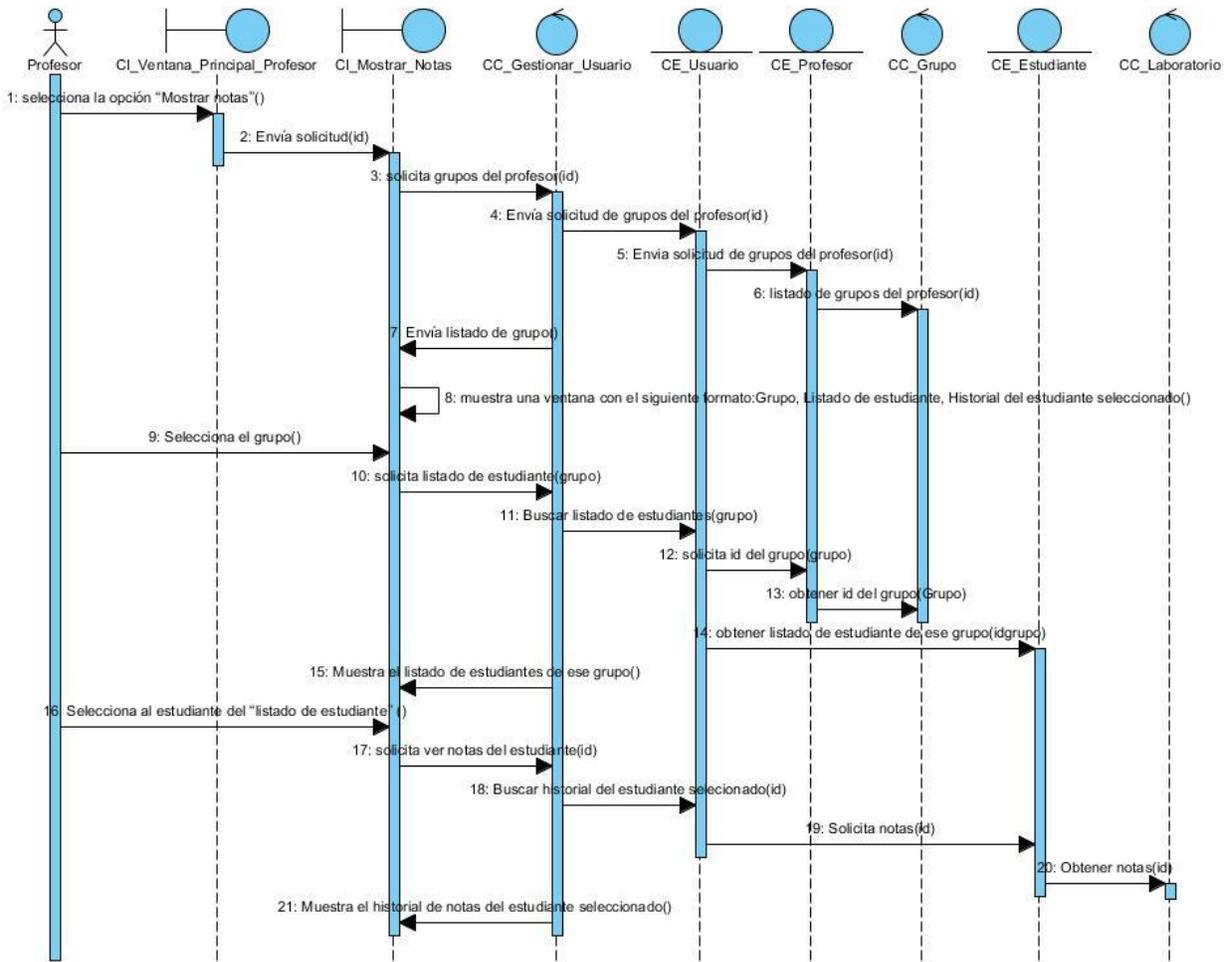


Figura 54. Diagrama de secuencia Mostrar notas.

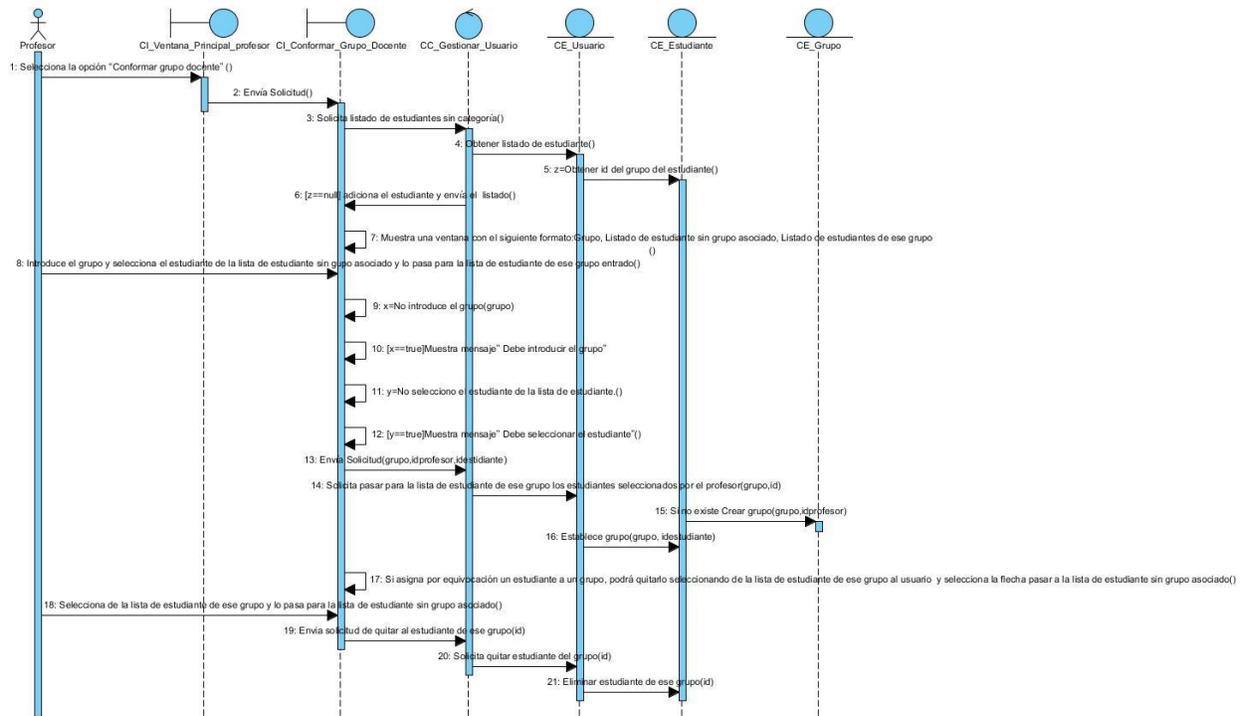


Figura 55. Diagrama de secuencia Conformar grupo docente.