



Facultad 5

ANÁLISIS Y DISEÑO DE UN LABORATORIO VIRTUAL DE ENTRENAMIENTO LAPAROSCÓPICO BÁSICO.

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

Autor(es): Sonia Beatriz Franco Kile

Tutor(es): Ing. Sailyn Salas Hechavarria

Co-tutor(es): Ing. Roxana Martínez Ferrer

“Año 54 del Triunfo de la Revolución”

La Habana, Cuba

Junio 2012

“Seamos realistas, soñemos lo imposible, porque de lo posible ya se ha hablado demasiado”.

Ernesto Che Guevara

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 _____.

Sonia Beatriz Franco Kile

Ing. Saily Salas Hechavarría

DATOS DE CONTACTO

Ing. Sailyn Salas Hechavarría: Graduada de Ingeniera en Ciencias Informáticas en el año 2008 y profesora instructora con cuatro años de experiencia docente en la Universidad de Ciencias Informáticas y nueve en la producción de software, e-mail: ssalas@uci.cu.

AGRADECIMIENTOS

A mis abuelos, quienes siempre me apoyaron con su espíritu alentador, contribuyendo incondicionalmente a lograr mis metas y objetivos propuestos.

A mis padres, por brindarme la fuerza necesaria para continuar y por ayudarme en lo que fuera posible, dándome consejos y orientación.

A Tatica, por mimarme tanto y por sus consejos.

A mi familia, por su constante apoyo, gracias por hacerme quien soy.

A Lisandra, mi amiga imperecedera, gracias por aguantarme todos estos años.

A Yuniel, por ser la brújula que me conduce hacia las más insólitas aventuras.

A los amigos que me ha dado la UCI, Leyíta, Titi, Bell, Luanner, Osmel, Jklo, José Ángel, Hamler y Arlety, a todos gracias por compartir conmigo un poquito de su esplendor.

Al Dr. Miguel González Sabin, por su tiempo y dedicación.

A Josito, por su permanente disposición y desinteresada ayuda.

A mis tutoras y a todos los que me ayudaron, Yeily, Yaíma, Belkis, Alejandro y Orlay gracias por sus valiosas sugerencias y acertados aportes durante el desarrollo de este trabajo.

A todas las personas que generosamente me dieron su apoyo en beneficio a mi desarrollo profesional y personal.

DEDICATORIA

A mis abuelos, para compensar todo el empeño y la dedicación que han puesto en mí.

A mis padres, por ser un ejemplo a seguir.

A Tatica, por su infinita bondad y amor.

A mi familia, porque siempre he contado con ellos para todo.

A Lisandra, por apoyarme incondicionalmente.

A mi nene precioso, por ser la luz que alumbró mi universo.

A mis amigos de la UCI, por todo lo que he aprendido junto a ellos en los momentos felices y tristes que hemos compartido.

RESUMEN

Durante las últimas décadas se ha experimentado un extraordinario progreso en el desarrollo tecnológico. Uno de los sectores de la sociedad sobre el que ha impactado significativamente es la salud. Las Ciencias Médicas en Cuba no están ajenas a este desarrollo lo cual ha propiciado el surgimiento de nuevos escenarios docentes y la utilización de nuevos medios de enseñanza, desde las videoconferencias, teleconferencias, aulas virtuales hasta sistemas de programas informáticos educativos.

Hoy en día las técnicas de cirugía mínimamente invasivas se utilizan cada vez más como una alternativa a un procedimiento quirúrgico de cirugía abierta, pues en este tipo de intervenciones se disminuyen los riesgos quirúrgicos. Sin embargo, la destreza y habilidad que los cirujanos requieren para realizar tales procedimientos es alta y las metodologías de entrenamiento no son siempre las apropiadas pues el entrenamiento en animales o en la sala de cirugía con demostraciones prácticas posee ciertas desventajas como costos elevados e implicaciones éticas. Por lo que el objetivo de este trabajo es diseñar prácticas de laboratorios virtuales para la adquisición de técnicas quirúrgicas laparoscópicas, con el fin de propiciar ambientes de aprendizaje alternativos que promuevan el desarrollo de habilidades quirúrgicas permitiendo a los estudiantes experimentar sin riesgo alguno y repetir la práctica tantas veces como sea necesario.

Palabras clave: diseño, habilidades laparoscópicas, laboratorio virtual

ÍNDICE DE CONTENIDOS

INTRODUCCIÓN	1
CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA	5
Introducción	5
1.1 Laboratorios Virtuales	5
1.1.1 Definiciones acerca de los laboratorios virtuales	5
1.3 Ventajas y desventajas de un laboratorio virtual	6
1.3.1 Ventajas	6
1.3.2 Desventajas.....	7
1.4 Cirugía de mínimo acceso	7
1.5 Simuladores quirúrgicos virtuales	8
1.5.1 Generaciones de simuladores quirúrgicos virtuales	8
1.5.2 Simuladores virtuales cirugía de mínimo acceso	10
1.5.3 Necesidad de un laboratorio virtual de cirugía de mínimo acceso	11
1.6 Metodologías de desarrollo de software	12
1.6.1 Metodologías pesadas	12
1.6.2 Metodologías ágiles	13
1.7 Lenguaje de modelado	14
1.8 Herramientas CASE	15
1.8.1 Rational Rose	15
1.8.2 Visual Paradigm.....	16
1.9 Lenguaje de programación	16
1.10 Motor gráfico	17
1.11 Entorno de desarrollo integrado	18
1.11.1 Eclipse	18
1.11.2 Qt Creator.....	19
Consideraciones parciales	20
CAPÍTULO 2. ANÁLISIS Y DISEÑO DEL SISTEMA	21
Introducción	21
2.1 Características del sistema	21
2.1.1 Modelo de negocio	21

2.1.2 Descripción del negocio	21
2.1.3 Reglas del negocio.....	22
2.1.4 Modelo de Casos de Uso del negocio	22
2.1.4.1 Actores del negocio	22
2.1.4.2 Casos de uso del negocio	23
2.1.4.3 Diagrama de casos de uso del negocio	23
2.1.4.4 Especificación de los casos de uso del negocio.....	23
2.1.4.5 Diagrama de actividades.....	24
2.1.4.6 Trabajador del negocio	25
2.1.4.7 Modelo de objetos del negocio	25
2.1.5 Especificación de requisitos	26
2.1.5.1 Requisitos funcionales	26
2.1.5.2 Requisitos no funcionales	27
2.1.6 Modelo del sistema	28
1.1.6.1 Actores del sistema.....	28
2.1.6.2 Diagrama de casos de uso del sistema.....	29
2.1.6.3 Especificación de casos de uso	30
2.2 Análisis y diseño del sistema.....	43
2.2.1 Estilos arquitectónicos.....	44
2.2.1.1 Estilo basado en capas.....	44
2.2.1.2 Estilo basado en componentes	44
2.2.3 Patrones de diseño	45
2.2.3.1 Patrones de diseño aplicados.....	45
2.2.4 Diagrama de clases del análisis.....	46
2.2.5 Diagrama de interacción	47
2.2.5.1 Diagramas de colaboración	48
2.2.6 Diagrama de clases del diseño	53
2.2.6.1 Diagrama de secuencia del diseño.....	55
2.2.7 Diagrama de Despliegue.....	57
Consideraciones parciales.....	57
<i>CAPÍTULO 3. VALIDACIÓN DE LA PROPUESTA DE DISEÑO</i>	<i>58</i>
Introducción.....	58
3.1 Validación con prototipos.....	58
3.1.1 Propósitos del prototipo.....	58
3.1.2 Clases de prototipo	58
3.1.3 Prototipo de características seleccionadas	59
Consideraciones parciales.....	61

CONCLUSIONES GENERALES.....	62
RECOMENDACIONES.....	63
REFERENCIAS BIBLIOGRÁFICAS.....	64

ÍNDICE DE FIGURAS

Figura 1. Distribución cronológica de los simuladores quirúrgicos	9
Figura 2. Actor del Negocio	22
Figura 3. Caso de uso del negocio.....	23
Figura 4. Diagrama de casos de uso del negocio.....	23
Figura 5. Diagrama de actividades.....	25
Figura 6. Diagrama de clases del modelo de objetos del negocio.....	26
Figura 7. Diagrama de casos de uso del sistema.....	29
Figura 8. Estilo basado en capas	44
Figura 9. Arquitectura del laboratorio virtual	45
Figura 10. Diagrama de clases del análisis Registrar usuario	46
Figura 11. Diagrama de clases del análisis Seleccionar y mostrar objeto de la práctica	46
Figura 12. Diagrama de clases del análisis Administrar objeto	47
Figura 13. Diagrama de clases del análisis Realizar práctica.....	47
Figura 14. Diagrama de clases del análisis Generar reporte	47
Figura 15. Diagrama de colaboración Registrar usuario	48
Figura 16. Diagrama de colaboración Seleccionar y mostrar objeto de la práctica.....	48
Figura 17. Diagrama de colaboración Realizar práctica sección Ejercicio 1	48
Figura 18. Diagrama de colaboración Realizar práctica sección Ejercicio 2	49
Figura 19. Diagrama de colaboración Realizar práctica sección Ejercicio 3	49
Figura 20. Diagrama de colaboración Realizar práctica sección Ejercicio 4	50
Figura 21. Diagrama de colaboración Realizar práctica sección Ejercicio 5	50
Figura 22. Diagrama de colaboración Realizar práctica sección Ejercicio 6	51
Figura 23. Diagrama de colaboración Realizar práctica sección Ejercicio 7	51
Figura 24. Diagrama de colaboración Realizar práctica sección Ejercicio 8	52
Figura 25. Diagrama de colaboración Generar reporte	52
Figura 26. Diagrama de colaboración Consultar reporte	53
Figura 27. Paquetes de las clases del diseño	53
Figura 28. Capa de presentación	54
Figura 29. Capa Lógica.....	54
Figura 30. Capa Soporte.....	55
Figura 31. Diagrama de secuencia Realizar práctica	55
Figura 32. Diagrama de secuencia Generar reporte	56
Figura 33. Diagrama de secuencia Consultar reporte.....	56
Figura 34. Diagrama de despliegue	57
Figura 35. RF1 Registrar usuario	59
Figura 36. RF2 Seleccionar objetos.....	60
Figura 37. RF3 Mostrar objetos	60
Figura 38. RF4.2 Eliminar objetos	61

ÍNDICE DE TABLAS

Tabla 1. Simuladores de cirugía de mínimo acceso.....	10
Tabla 2. Especificación de casos de uso del negocio.....	24
Tabla 3. Descripción de los actores del sistema.....	29
Tabla 4. CUS1 Registrar usuario.....	31
Tabla 5. CUS2 Seleccionar objeto de la práctica.....	32
Tabla 6. CUS3 Mostrar objeto de la práctica.....	33
Tabla 7. CUS4 Administrar objeto.....	35
Tabla 8. CUS5 Realizar práctica.....	41
Tabla 9. CUS6 Generar reporte.....	42
Tabla 10. CUS7 Consultar reporte.....	43

INTRODUCCIÓN

El progreso de la revolución informática unido al auge de las Tecnologías de la Información y las Comunicaciones (TIC) son incuestionables, forman parte de la cultura tecnológica que nos rodea y con la cual debemos convivir pues amplían nuestras capacidades físicas, mentales así como las posibilidades de desarrollo social.

Entre las ramas sociales, las más favorecidas son la educación y la salud, las que tienen la posibilidad de poner a disposición de los educandos y de la sociedad en general, por medio de las TIC, un enorme compendio de contenidos que enriquecen las ciencias generales y específicas, propiciando un mayor acceso, por tanto, una mejor difusión de la información.

Las Ciencias Médicas en Cuba no están ajenas a este desarrollo lo cual ha propiciado el surgimiento de nuevos escenarios docentes y la utilización de nuevos medios de enseñanza, desde las videoconferencias, teleconferencias, aulas virtuales, sistemas de programas informáticos educativos, multimedia y cursos a distancia, todo lo cual posibilita la adquisición de habilidades.

Algunos de los medios de enseñanza utilizados son los siguientes:

1. Cajas de entrenamiento: Permite dominar el instrumental y ejercitar gestos quirúrgicos (suturas, nudos)
2. Prácticas con animales de experimentación: Se consigue una experiencia real en maniobras quirúrgicas y en la realización de intervenciones completas.
3. Demostraciones prácticas: Permite apreciar detalles técnicos y aprender soluciones a situaciones imprevistas y complicaciones surgidas durante su realización.
4. Herramientas virtuales: Brinda la posibilidad de intentar y fallar sin consecuencias para el paciente, pero ello requiere que los objetos y órganos internos del paciente se visualicen de la forma más realista posible, y que estos respondan y en tiempo real, a las interacciones de los usuarios.

Hoy en día las técnicas de cirugía mínimamente invasivas se utilizan cada vez más como una alternativa a un procedimiento quirúrgico de cirugía abierta, pues en este tipo de intervenciones se disminuyen los riesgos quirúrgicos, se acortan las incapacidades post-operatorias y se reducen el tiempo de

hospitalización al igual que las posibles secuelas físicas. Sin embargo la destreza y habilidad que los cirujanos requieren para realizar tales procedimientos es alta y las metodologías de entrenamiento no son siempre las apropiadas pues el entrenamiento en animales o en la sala de cirugía con demostraciones prácticas posee ciertas desventajas como costos elevados e implicaciones éticas.

Gracias al desarrollo de las TIC se han realizado un gran número de simuladores quirúrgicos en el mundo que simulan acciones típicas quirúrgicas como cauterización, corte y sutura pero su precio en el mercado es elevado y son de difícil acceso.

Atendiendo a todo ello la **situación problémica** es la siguiente:

Actualmente en nuestro país no se cuenta con prácticas de laboratorio virtual que permita llevar a cabo el entrenamiento de técnicas laparoscópicas lo cual:

- No permite simular situaciones que en la vida real los estudiantes tienen escasas posibilidades de realizar.
- Restringe la posibilidad de repetir prácticas las veces que se requiera.
- No admite a un número mayor de estudiantes experimentar con un laboratorio.
- No permite que los estudiantes aprendan por cuenta propia fomentando la capacidad de análisis, el pensamiento crítico y la utilización de la tecnología informática.
- Frena el aprendizaje constructivista.
- Imposibilita la disminución del costo de instalación y mantenimiento de un laboratorio convencional.

Dada la situación problémica previamente expuesta se define el siguiente **problema científico**: ¿Cómo simular el entrenamiento de técnicas laparoscópicas en escenas virtuales para facilitar su aprendizaje a menores costos?

Lo que determina como **objeto de estudio**: Diseño de prácticas de laboratorios virtuales para la medicina.

Luego, para darle solución al problema de investigación planteado se propone como **objetivo general**: Diseñar prácticas de laboratorios virtuales para la adquisición de técnicas quirúrgicas laparoscópicas.

Por todo lo anterior, se precisa como **campo de acción:** Diseño de prácticas de laboratorios virtuales para la adquisición de técnicas laparoscópicas.

Para darle solución al objetivo se proponen como **tareas de la investigación:**

1. Elaboración del marco teórico de la investigación a partir del estudio del estado del arte existente sobre el tema en la actualidad.
2. Selección de las técnicas laparoscópicas para definir los ejercicios a desarrollar en el sistema.
3. Obtención de los requerimientos funcionales y no funcionales para modelar las funcionalidades y restricciones que tiene el sistema a diseñar.
4. Elaboración del modelo de diseño para plasmar cómo debe quedar el sistema.
5. Validación del diseño que se propone a través de un prototipo de características seleccionadas.

Entre los métodos de investigación científica existentes fueron utilizados los siguientes:

De los **métodos teóricos** se utilizó:

Método de Análisis y Síntesis: Principalmente en la precisión de los fundamentos teóricos relacionados con los laboratorios virtuales para apoyar el proceso de enseñanza-aprendizaje en la medicina.

Método de Análisis histórico-lógico: Para profundizar en los antecedentes del desarrollo de este tipo de software a nivel nacional e internacional y en la evolución de los mismos.

De los **métodos empíricos** se utilizó:

Entrevista: Para recopilar las características más importantes de los procesos del negocio al cual responden los laboratorios virtuales de cirugía.

Consulta de especialistas: Para indagar sobre la calidad de lo realizado.

Consulta de clientes: Para indagar sobre la aceptación de lo realizado.

Con la realización del trabajo se defiende la siguiente **idea:**

El diseño de laboratorio virtual propuesto contribuirá a la adquisición de habilidades laparoscópicas y reducirá el costo de instalación y mantenimiento de un laboratorio convencional.

El contenido se estructura de la siguiente manera:

Capítulo 1: Fundamentación teórica.

Se expone el marco conceptual de la propuesta de solución. Se realiza un estudio de las herramientas y tecnologías a utilizar en el desarrollo del sistema.

Capítulo 2: Análisis y Diseño del sistema.

Se identifican los procesos a automatizar, se describen de los requisitos funcionales y no funcionales y se elabora el modelo de diseño.

Capítulo 3: Validación de la solución propuesta.

Se valida la solución propuesta mediante un prototipo de características seleccionadas.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

Introducción

El presente capítulo abarca el estado del arte. Se elabora el marco teórico de la investigación para lo cual se estudian definiciones sobre los laboratorios virtuales, se caracteriza la evolución de los mismos al conocer los antecedentes históricos que los originaron, analizando también sus ventajas y desventajas. Se profundiza además sobre el surgimiento de la cirugía de mínimo acceso y los simuladores quirúrgicos para este tipo de cirugía. Se realiza además un análisis de las herramientas y tecnologías a utilizar en el análisis y diseño del sistema. En dependencia de las características de cada una se opta por la más adecuada.

1.1 Laboratorios Virtuales

Entre las herramientas virtuales usadas en la medicina se puede mencionar el laboratorio virtual (LV), que gracias al desarrollo de las TIC se han convertido en un instrumento para la mejora del proceso de enseñanza-aprendizaje en los estudiantes. A continuación se ofrecen algunas de las definiciones más importantes, mostrando claramente las características esenciales de los LV.

1.1.1 Definiciones acerca de los laboratorios virtuales

Varios especialistas del tema han dado disímiles definiciones de los LV. Según el concepto dado en la reunión de expertos en LV efectuada en París en el 2000 un LV es “un espacio electrónico de trabajo concebido para la colaboración y la experimentación a distancia con objeto de investigar o realizar otras actividades creativas, elaborar y difundir resultados mediante tecnologías difundidas de información y comunicación”.(1)

Otra definición plantea:

“Un laboratorio virtual es una herramienta multimedia interactiva que sirve para mejorar y complementar el proceso de aprendizaje. Un laboratorio virtual tiene las siguientes características principales:

- Tiene una interfaz de usuario intuitivo y fácil de utilizar.

- Utiliza instrumentación simulada interactiva que posee una funcionalidad similar a la de los instrumentos reales.
- Relaciona los conceptos prácticos con los teóricos mediante un conjunto de experimentos adecuadamente diseñados.
- Es fácilmente ampliable“. (2).

También se dice que un LV:

“Es un instrumento diseñado para aprender a resolver problemas experimentales y puede ser de gran utilidad como un recurso complementario a las prácticas de laboratorio. La principal finalidad del laboratorio virtual es la de ser un instrumento útil para aprender a diseñar experimentos e integrar conceptos teóricos y prácticos de diferentes asignaturas “. (3)

A partir de las definiciones anteriores se considera que un LV no es más que: un simulador de prácticas de laboratorio tradicionales dentro de una computadora, es decir virtual, lo cual lo convierte en una herramienta diseñada para mejorar el proceso de aprendizaje, para aprender y ejercitar técnicas así como integrar conceptos teóricos y prácticos de diferentes asignaturas.

1.3 Ventajas y desventajas de un laboratorio virtual

1.3.1 Ventajas

Son vastas las ventajas de usar un LV, a continuación se evidencian algunas:

1. “Acerca y facilita a un mayor número de estudiantes y especialistas la realización de experiencias, aunque estudiante y laboratorio no coincidan en el espacio. El estudiante accede a los equipos del laboratorio a través de una interfaz, pudiendo experimentar sin riesgo alguno.
2. Reduce el costo del montaje y mantenimiento de los laboratorios tradicionales, siendo una alternativa barata y eficiente.
3. Los estudiantes aprenden mediante prueba y error, mientras que especialistas se ejercitan sin miedo a provocar un accidente.

4. Se convierte en una ayuda interactiva para el aprendizaje de contenidos difíciles de demostrar en la realidad.
5. Se incorporan las TIC en las prácticas educativas y sociales para beneficio de los estudiantes.” (4).

1.3.2 Desventajas

El uso de LV trae consigo también desventajas:

1. “No puede sustituir la experiencia práctica altamente enriquecedora del laboratorio tradicional.
2. Se corre el riesgo de que el estudiante se comporte como un mero espectador. Es necesario que el estudiante realice una actividad ordenada y progresiva, conducente a alcanzar objetivos básicos concretos.
3. No todas las instituciones educativas cuentan con un área de desarrollo de software de apoyo académico, que den soporte al diseño e instalación del Laboratorio Virtual” (4).

1.4 Cirugía de mínimo acceso

La cirugía ha sufrido un cambio profundo, una verdadera revolución, con el advenimiento de la cirugía de acceso mínimo.

Existen muchos procedimientos en la cirugía laparoscópica que son nuevos en cirugía general, entre ellos: acceso de los trócares a través de la pared abdominal, manipulación de instrumentos miniaturizados con mango largo, la pérdida del tacto por inhabilidad de tocar los órganos directamente, trabajo sin visión directa y sin orientación tridimensional.

“La cirugía de mínimo acceso tuvo su origen en Francia, en 1986, iniciada por un médico nombrado Philipe Mouret. Luego se extendió hacia los Estados Unidos y Europa. En Cuba se comenzaron las operaciones de mínimo acceso en 1991 y estuvo aplicándose sólo en la capital hasta el 11 de marzo de 1996, fecha en que se realizó la primera operación de este tipo en el Hospital General Juan Bruno Zayas Alfonso, de Santiago de Cuba.” (5)

“Entre las ventajas de la cirugía de mínimo acceso, están en primer lugar, el aspecto humano al producir menos dolor. También la significativa reducción de complicaciones de las heridas, por su tamaño pequeño, además del beneficio estético- sobre todo en las mujeres- al dejar solo pequeñas cicatrices que

con el tiempo desaparecen. También está el beneficio económico para las personas y para la economía del país, pues los pacientes pueden reintegrarse de forma rápida a la sociedad y a su labor, a la vez que los hospitales ahorran recursos al reducir la estadía de los pacientes, cuyo costo es de unos ciento sesenta pesos por día.” (5)

1.5 Simuladores quirúrgicos virtuales

“En los últimos años el desarrollo de la informática gráfica ha permitido ampliar los entornos visuales que permiten el entrenamiento de determinadas técnicas quirúrgicas en un ambiente virtual. Un buen simulador quirúrgico debe de cumplir una serie de características: debe mostrar los órganos internos de manera realista, debe garantizar que éstos respondan a las interacciones con el cirujano, por ejemplo deformándose y que respondan mediante modificaciones estructurales a acciones quirúrgicas habituales como cortes, cauterización o sutura” (6).

1.5.1 Generaciones de simuladores quirúrgicos virtuales

En función de que consigan estos objetivos los simuladores quirúrgicos virtuales se agrupan en tres generaciones:

- Primera generación: Únicamente consideran la naturaleza geométrica de las estructuras anatómicas, pero no permiten interactuar con ellos.
- Segunda generación: Permiten una cierta interacción con las estructuras anatómicas.
- Tercera generación: Permiten una interacción con las estructuras y además tienen en cuenta la naturaleza funcional de los órganos.

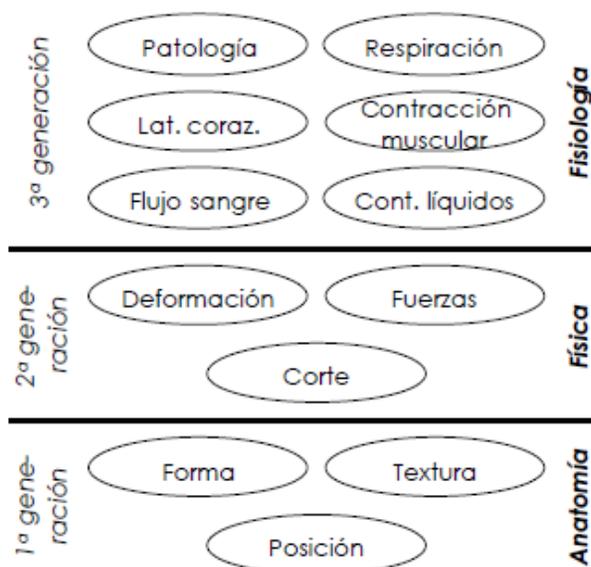


Figura 1. Distribución cronológica de los simuladores quirúrgicos (6)

Actualmente la mayoría de los simuladores virtuales están incluidos en los de segunda generación, para la cual existen cuatro niveles de complejidad:

1. “Simuladores de aguja que disponen de objetos visuales simples con un mínimo grado de libertad.
2. Simuladores que permiten una exploración endoscópica o instalación de catéteres.
3. Simuladores para el entrenamiento en una tarea, que disponen de uno o dos instrumentos que interaccionen con el entorno virtual.
4. Simuladores completos: permiten el entrenamiento en determinados procedimientos quirúrgicos laparoscópicos, artroscopia, ginecología o cirugía intraocular.” (7).

1.5.2 Simuladores virtuales cirugía de mínimo acceso

Simulador	Generación	Nivel de complejidad	Fabricante	Características
Eva	Primera	-	Pro-Delphus	Permite los siguientes entrenamientos quirúrgicos simulados: triangulación quirúrgica, entrenamiento de la visión endoscópica, Miomectomia, Salpingectomía, Ovarioplastia, enseñanza de la sutura endoscópica (intestinal, vejiga, útero, uréter).
LAP Mentor	Segunda	Cuarto	Simbionix USA Corporation	Es el sistema más desarrollado tecnológicamente a nivel mundial, permite entrenar desde habilidades básicas de laparoscopia y suturas, hasta la manipulación de las pinzas en procedimientos de colecistectomía, reparación de hernia, ginecológicos, urológicos, bariátricos, colorrectales y nefrectomía.
Simulador para laparoscopia	Segunda	Cuarto	Medical Simulator	Se compone de un hardware que incorpora diferentes programas y una pantalla en la que se ve la imagen en dos dimensiones. Este hardware interacciona con los mandos (dos pinzas fijas) del software que como principal característica tiene feed-back de fuerza.

Tabla 1. Simuladores de cirugía de mínimo acceso.

“SimuLab Corporation también ofrece simuladores quirúrgicos y entrenadores de tareas para la comunidad educativa médica en todo el mundo. Entre sus productos destacan varios simuladores básicos para laparoscopia como el LapTrainer Skills Set #1 por el precio de doscientos veinticinco dólares o el LapTrainer con SimuVision por el valor de mil setecientos noventa y cinco dólares.” (8)

Como se pudo apreciar existen varios simuladores para entrenar la adquisición y especialización de técnicas para realizar cirugías de mínimo acceso, pero son de difícil acceso al ser desarrollados por compañías privadas, así como por tener un elevado precio en el mercado.

1.5.3 Necesidad de un laboratorio virtual de cirugía de mínimo acceso

“Las necesidades de la sociedad, el desarrollo científico técnico y la competencia y desempeño profesional en las especialidades quirúrgicas nos pone ante otros cuestionamientos: ¿Se puede aprender la cirugía laparoscópica solo de mirar?; ¿las habilidades que deben adquirir los educandos son otras?; ¿los programas académicos y la metodología hasta ahora aplicada son los adecuados para el aprendizaje de los procedimientos laparoscópicos? y, sobre todo, ¿cómo enseñar y aprender estas técnicas, si también son nuevas para los profesores?” (9)

“Durante estos años y en el futuro, la responsabilidad de asegurar el adecuado entrenamiento de los cirujanos en formación y el perfeccionamiento de los que ya poseen un nivel básico en cirugía debe recaer en las instituciones dedicadas a la formación de especialidades quirúrgicas, rectoradas por la universidad médica y el ministerio de salud pública.” (9)

“El advenimiento de la cirugía laparoscópica o de mínimo acceso y su masificación ha puesto a prueba el sistema clásico de enseñanza de la cirugía, puesto que a la transmisión de conocimientos sobre una técnica quirúrgica se ha añadido la adquisición de nuevas habilidades, como la coordinación ojo-mano, la visualización del campo quirúrgico en monitores, la ausencia de la tercera dimensión y el desarrollo de una nueva forma de sentir el tacto.” (9)

Uno de los métodos utilizados para la superación profesional en el posgrado en cirugía laparoscópica son los centros de entrenamiento. En Cuba en el año 1993 se funda el Centro Nacional de Cirugía de Mínimo Acceso, una institución asistencial, docente e investigativa que tiene entre sus misiones la formación posgraduada y el perfeccionamiento profesional. Para contribuir a estos fines en el año 2002 se inaugura

el Centro de Entrenamiento, el cual se utiliza como escenario docente de los programas de superación profesional que son parte de la formación quirúrgica continuada que se oferta a los cirujanos, urólogos, ginecólogos y otros profesionales de especialidades afines, tanto cubanos, como latinoamericanos y de otros países.

Hoy más del 50% de las enfermedades que tienen como tratamiento la cirugía, se tratan por mínimo acceso. Esta técnica quirúrgica abarca nueve especialidades, entre ellas gastroenterología, urología, ginecología, cirugía pediátrica, reumatología y ortopedia.

Aunque en el Centro de Entrenamiento existe un simulador *Pelvictrainer*, y se usan otros métodos como la práctica con animales o la observación a expertos, estas acciones no son suficientes para entrenar la cantidad de especialistas que realizan intervenciones de mínimo acceso.

Se plantea, por consiguiente, el diseño de un LV para entrenar técnicas laparoscópicas como la respuesta a la necesidad de atenuar el problema pedagógico de preparación de estudiantes en el aprendizaje y perfeccionamiento de este tipo de procedimiento.

1.6 Metodologías de desarrollo de software

Las metodologías de desarrollo de software surgen ante la necesidad de utilizar una serie de procedimientos, técnicas, herramientas y soporte documental a la hora de desarrollar un producto software. Se clasifican en dos grandes grupos:

- “Las metodologías orientadas al control de los procesos, estableciendo rigurosamente las actividades a desarrollar, herramientas a utilizar y notaciones que se usarán. Estas metodologías son llamadas metodologías tradicionales o pesadas.
- Las metodologías orientadas a la interacción con el cliente y el desarrollo incremental del software, mostrando versiones parcialmente funcionales del software al cliente en intervalos cortos de tiempo, para que pueda evaluar y sugerir cambios en el producto según se va desarrollando. Estas son llamadas metodologías ágiles o ligeras.” (10)

1.6.1 Metodologías pesadas

Entre las metodologías tradicionales o pesadas se pueden citar:

- “RUP (Rational Unified Process): Proceso de desarrollo de software que junto al UML (Unified Modeling Language), constituye la metodología estándar más utilizada para el análisis, implementación y

documentación de sistemas orientados a objetos. RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

RUP como metodología que tiene tres características importantes que la hacen única, está dirigida por casos de uso, centrada en la arquitectura, iterativa e incremental. Otras de sus características más importantes son:

- Propone un flujo de trabajo específicamente para el Análisis y Diseño del sistema, donde se especifican los requisitos identificados y se describen en términos de diseño sobre cómo se va a implementar el sistema.
 - Provee vasta documentación al equipo de trabajo para documentar bien el software (planillas).
 - Describe como obtener, organizar y documentar todas las funcionalidades y restricciones del sistema.
 - En cada una de sus fases obtiene una serie de artefactos que sirven para comprender mejor tanto el Análisis como el Diseño del sistema.
- MSF (Microsoft Solution Framework): Es un conjunto de principios, modelos, disciplinas, conceptos y directrices para la entrega de soluciones de tecnología de la información de Microsoft. MSF no se limita al desarrollo de aplicaciones sólo, es también aplicable a otros proyectos como los proyectos de despliegue de redes o infraestructura. MSF no obliga al desarrollador a utilizar una metodología específica, pero les permite decidir qué método utilizar.” (10)

1.6.2 Metodologías ágiles

“En febrero de 2001, tras una reunión celebrada en Utah-EEUU, nace el término “ágil” aplicado al desarrollo de software. En esta reunión participan un grupo de diecisiete expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Tras esta reunión se creó The Agile Alliance 3, una organización, sin ánimo de lucro, dedicada a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida fue el Manifiesto Ágil, un documento que resume la filosofía “ágil”. (11)

A continuación se dan características de metodologías ágiles que ya estaban siendo utilizadas con éxito en proyectos reales, pero les faltaba una mayor difusión y reconocimiento.

- XP(Extreme Programming): es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo. XP se basa en realimentación 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. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, donde existe un alto riesgo técnico.
- SCRUM: Desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos diez años. Está especialmente indicada para proyectos con un rápido cambio de requisitos. Sus principales características se pueden resumir en:
 1. El desarrollo de software se realiza mediante iteraciones, denominadas sprints, con una duración de treinta días. El resultado de cada sprint es un incremento ejecutable que se muestra al cliente.
 2. Reuniones a lo largo del proyecto, entre ellas destaca la reunión diaria de quince minutos del equipo de desarrollo para coordinación e integración.” (11)

No existe una metodología universal para hacer frente con éxito a cualquier proyecto de desarrollo de software. Toda metodología debe ser adaptada al contexto del proyecto (recursos técnicos y humanos, tiempo de desarrollo, tipo de sistema).

A partir de las metodologías de desarrollo de software analizadas se propone el uso de la metodología tradicional RUP pues es la metodología estándar más utilizada para el análisis, implementación y documentación en sistemas orientados a objetos.

1.7 Lenguaje de modelado

“El lenguaje para modelamiento unificado, UML por sus siglas en inglés, emergió en la década de los años noventa luego de la búsqueda de un lenguaje de modelamiento que unificara a la industria, que siguió a la "guerra de métodos" de las décadas anteriores. A pesar de que UML evolucionó primeramente de varios métodos orientados al objeto de segunda generación (en nivel de notación), UML no es simplemente un lenguaje para modelamiento orientado al objeto de tercera generación.

UML es un lenguaje de modelamiento para la especificación, visualización, construcción y documentación de los artefactos de un proceso de desarrollo de software:

- Como un lenguaje, es usado para la comunicación. Es decir, un medio para capturar el conocimiento (semánticas) respecto a un tema y expresar el conocimiento (sintaxis) resguardando el tema propósito de la comunicación. El tema es el proceso en estudio.
- Como un lenguaje para modelamiento, se enfoca en la comprensión de un tema a través de la formulación de un modelo del tema y su contexto respectivo.
- Para especificar sistemas, puede ser usado para comunicar "qué" se requiere de un sistema y "cómo" un sistema puede ser realizado.
- Puede ser usado para describir visualmente un sistema antes de ser realizado.
- Puede ser usado para guiar la realización de un sistema similar a los "planos".
- Puede ser usado para capturar conocimiento respecto a un sistema a lo largo de todo el proceso de su ciclo de vida.”(12)

1.8 Herramientas CASE

Se pueden definir a las herramientas CASE como un conjunto de programas que dan asistencia a los analistas, ingenieros de software y desarrolladores, durante todos los pasos del ciclo de vida de desarrollo de un software. Existen varias herramientas de este tipo, entre las más reconocidas y usadas se destacan Rational Rose y Visual Paradigm. A continuación se exponen sus principales características.

1.8.1 Rational Rose

“Rational Rose es una herramienta basada en el UML, que permite crear los diagramas que se van generando durante el proceso de ingeniería en el desarrollo del software.

Las personas que desarrollaron el Proceso Unificado del Rational (RUP) son miembros de Rational Corporation por lo tanto, Rational brinda muchas facilidades en la generación de la documentación del software que se esté desarrollando, además de que posee un gran número de estereotipos predefinidos que facilitan el proceso de modelación del software.

En la definición de sistemas, esta herramienta permite que el equipo de desarrollo entienda mejor el problema, que identifique las necesidades del cliente en forma más efectiva y comunique la solución

propuesta de forma más clara. Rational permite completar una gran parte de las disciplinas (flujos fundamentales) de RUP tales como:

- Captura de requisitos (parcialmente).
- Análisis y diseño (completamente).
- Implementación (como ayuda).
- Control de cambios y gestión de configuración (parcialmente).”(13)

1.8.2 Visual Paradigm

“Visual Paradigm ha sido concebida para soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de todo tipo de diagramas. Constituye una herramienta de software libre de probada utilidad para el analista.

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.

Se caracteriza por:

- Software libre.
- Disponibilidad en múltiples plataformas (Windows, Linux).
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería directa e inversa.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Licencia: gratuita y comercial.
- Fácil de instalar y actualizar.
- Soporte de UML versión 2.1.”(14)

Luego de haber realizado un profundo análisis de las herramientas estudiadas, se selecciona la herramienta CASE Visual Paradigm por las potencialidades que ofrece.

1.9 Lenguaje de programación

Un lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una computadora. Consiste en un conjunto de reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos, respectivamente. Puede clasificarse teniendo en cuenta distintos criterios:

lenguajes interpretados, lenguajes compilados, lenguajes de script, entre otros. Estas características dividen a los lenguajes de programación en dos grupos, declarativos e imperativos. Para desarrollar aplicaciones de realidad virtual, existen en el mundo un conjunto de lenguajes de programación tanto para plataformas GNU/Linux, como para Windows, entre los que se destaca el lenguaje C++.

“C++ es un lenguaje de programación diseñado a mediados de los años 1980 por Bjarne Stroustrup. La intención de su creación fue el extender al exitoso lenguaje de programación C con mecanismos que permitan la manipulación de objetos. En ese sentido, desde el punto de vista de los lenguajes orientados a objetos, el C++ es un lenguaje híbrido.

Posteriormente se añadieron facilidades de programación genérica, que se sumó a los otros dos paradigmas que ya estaban admitidos (programación estructurada y la programación orientada a objetos). Por esto se suele decir que el C++ es un lenguaje de programación multiparadigma. Presenta además las siguientes características:

- está estandarizado por un comité de la Organización Internacional para la Estandarización (ISO).
- es un lenguaje compilado, compila directamente al código nativo de la máquina, lo que le permite ser uno de los lenguajes más rápidos del mundo, si es optimizado.
- es compatible con la comprobación de tipo estático y dinámico.
- permite conversiones de tipos a verificar, en tiempo de compilación o en tiempo de ejecución.”(15)

1.10 Motor gráfico

Un motor gráfico es un programa que controla, gestiona y actualiza gráficos 3D en tiempo real. Los motores gráficos facilitan la programación de juegos y aplicaciones de diseño sin tener que enfrentarse desde cero al lenguaje de programación utilizado en cuestión.

Ogre (Object Oriented Graphics Engine) es un motor multiplataforma de gráficos en 3D. La principal ventaja de Ogre sobre otros motores 3D es que es un proyecto de código abierto bajo licencia LGPL, lo cual significa que su uso es gratuito y solo existen unas pocas exigencias para su uso.

Ogre está diseñado desde el principio con la idea de la orientación a objetos, por lo que su interfaz es clara, intuitiva y fácil de usar. “Algunas de sus características de productividad son:

- Es orientado a objetos, diseñado para minimizar el esfuerzo necesario para dibujar escenas 3D.

- Posee requisitos comunes como la gestión de estado de representación, que se realizan de forma automática ahorrando un tiempo valioso.
- Ofrece un diseño limpio, ordenado y documentación completa de todas las clases de motores.
- Es además un motor probado y estable utilizado en varios productos comerciales".(16)

1.11 Entorno de desarrollo integrado

Un entorno de desarrollo integrado llamado también IDE (Integrated Development Environment), es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un solo lenguaje de programación o bien puede utilizarse para varios.

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Los IDE pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes.

Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, PHP, Python, Java, C#, Delphi, Visual Basic, etc.

1.11.1 Eclipse

"Eclipse es un IDE de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores.

Es también una comunidad de usuarios, extendiendo constantemente las áreas de aplicación cubiertas. Un ejemplo es el recientemente creado Eclipse Modeling Project. Fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge. Eclipse es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios.

Eclipse fue liberado originalmente bajo la Licencia Pública Común, pero después fue re-licenciado bajo la Licencia Pública Eclipse. La Fundación de Software Libre ha dicho que ambas licencias son licencias de software libre, pero son incompatibles con Licencia Pública General.

Emplea módulos (en inglés plug-in) para proporcionar toda su funcionalidad al frente de la plataforma de cliente enriquecido, a diferencia de otros entornos monolíticos donde las funcionalidades están todas

incluidas, las necesite o no el usuario. Este mecanismo de módulos es una plataforma ligera para componentes de software. Adicionalmente a permitirle a Eclipse extenderse usando otros lenguajes de programación como son C/C++ y Python, permite a Eclipse trabajar con lenguajes para procesado de texto como LaTeX, aplicaciones en red como Telnet y Sistema de gestión de base de datos.” (17)

1.11.2 Qt Creator

Qt Creator es un IDE desarrollado por TrollTech para el desarrollo de aplicaciones con bibliotecas de Qt, que es un framework muy útil para desarrollar aplicaciones de escritorio. “Entre sus principales características se pueden citar:

1. Se centra en proporcionar características que ayudan a los nuevos usuarios de Qt aprender y comenzar a desarrollar rápidamente.
2. Posee un resaltado auto-completado de código.
3. Se ajusta a las necesidades de los desarrolladores Qt.
4. Aumenta la productividad de los desarrolladores con experiencia en Qt.
5. El depurador visual 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.
6. Muestra la información en bruto procedente de GDB de una manera clara y concisa.
7. Es un entorno integrado para la creación y diseño de forms para proyectos C++ que permite diseñar rápidamente widgets y diálogos usando los mismos widgets que se usarán en su aplicación.
8. Está disponible para las plataformas: Linux, Mac OSX; Windows, Windows CE.
9. Posee un avanzado editor de código C++.
10. Es un entorno de desarrollo integrado software libre y de código abierto.” (18)

Después de realizar un análisis de los IDE caracterizados anteriormente, se considera que el más adecuado para el desarrollo del laboratorio virtual es Qt Creator, por las características que presenta.

Consideraciones parciales

- Se realizó un estudio sobre las definiciones de los laboratorios virtuales y el surgimiento de la cirugía de mínimo acceso.
- En el mercado internacional pueden encontrarse varios simuladores quirúrgicos para el entrenamiento de cirugía de mínimo acceso, pero son desarrollados por corporaciones privadas y poseen un elevado precio de mercado.
- Se estudiaron una serie de herramientas y tecnologías de las cuales se escogieron las más apropiadas para el desarrollo del laboratorio.

CAPÍTULO 2. ANÁLISIS Y DISEÑO DEL SISTEMA

Introducción

Este capítulo consta de dos partes fundamentales, primero se describe la solución propuesta y los procesos del negocio, así como la representación gráfica del modelo del negocio. Se plantean las reglas de negocio, los requisitos funcionales y no funcionales de la aplicación a desarrollar. Luego se elaboran los artefactos correspondientes a la fase de análisis y diseño de la solución que se propone.

2.1 Características del sistema

2.1.1 Modelo de negocio

En el proceso de desarrollo de software resulta útil la creación de modelos que organicen y muestren los detalles importantes de problemas reales que se vinculan con el sistema informático a construir. Estos modelos deben cumplir una serie de propiedades, entre ellas la de ser coherentes y relacionados. Uno de los modelos útiles previo al desarrollo de un software es el modelo del negocio.

“Los principales objetivos del modelamiento del negocio son:

- Comprender la estructura y la dinámica de la organización en la cual se va a implantar un sistema.
- Comprender los problemas actuales de la organización e identificar las mejoras potenciales.
- Asegurar que los consumidores, usuarios finales y desarrolladores tengan un entendimiento común de la organización.” (19).

2.1.2 Descripción del negocio

Los practicantes de ciencias médicas en Cuba necesitan prácticas de laboratorio que abarquen los temas y objetivos fundamentales de las disciplinas de cirugía de mínimo acceso, para desarrollar habilidades básicas en los estudiantes y lograr una mayor destreza en los especialistas.

Tradicionalmente en las universidades médicas se realizan otros tipos de entrenamiento como la práctica con animales o la observación a expertos, pero no son suficientes e implican riesgos éticos. Este LV les permitirá a los estudiantes realizar ejercicios de entrenamiento. Durante el desarrollo de la práctica el

usuario identificará cada uno de los elementos y componentes que conforman la misma, para desarrollarla correctamente.

2.1.3 Reglas del negocio

Las reglas de negocio se usan para regular algún aspecto del negocio pues describen políticas que deben cumplirse o condiciones que deben satisfacerse.

Se definieron las siguientes reglas del negocio:

1. La práctica solo la podrán realizar los estudiantes de las ciencias médicas.
2. El practicante no puede saltarse los ejercicios definidos para el laboratorio.
3. El practicante solo podrá ver su valoración cuando culmine la práctica de laboratorio.
4. El evaluador es el único que puede dar la valoración de todos los practicantes.

2.1.4 Modelo de Casos de Uso del negocio

“El modelo de casos de uso del negocio describe los procesos de un negocio (casos de uso del negocio) y su interacción con elementos externos (actores), tales como socios y clientes, es decir, describe las funciones que el negocio pretende realizar y su objetivo básico es describir cómo el negocio es utilizado por sus clientes y socios”. (19)

2.1.4.1 Actores del negocio

“Un actor del negocio es cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos; con los que el negocio interactúa. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados”. (19)

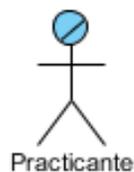


Figura 2. Actor del Negocio

2.1.4.2 Casos de uso del negocio

“Un caso de uso del negocio representa a un proceso de negocio, por lo que se corresponde con una secuencia de acciones que producen un resultado observable para ciertos actores del negocio”. (19).

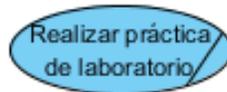


Figura 3. Caso de uso del negocio

2.1.4.3 Diagrama de casos de uso del negocio

“Un diagrama de casos de uso del negocio representa gráficamente a los procesos del negocio y su interacción con los actores del negocio.” (19).

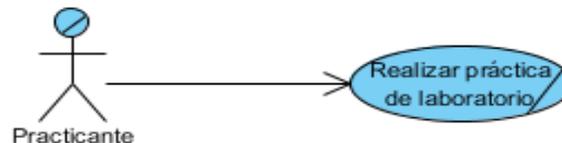


Figura 4. Diagrama de casos de uso del negocio

2.1.4.4 Especificación de los casos de uso del negocio

Nombre	Realizar práctica de laboratorio.
Actor del negocio:	Practicante (Inicia)
Propósito:	Realizar la práctica de laboratorio para entrenamiento laparoscópico.
Resumen:	El caso de uso inicia cuando el practicante empieza a realizar la práctica de laboratorio, de la cual recibe una valoración.
Flujo Normal de los eventos:	

Acciones del actor	Respuesta del proceso de negocio
1- Llega al laboratorio	1.1- El evaluador orienta la práctica.
2- Comienza a resolver la práctica de laboratorio.	
3- Culmina la práctica de laboratorio.	3.1- El evaluador valora la práctica de laboratorio. 3.2- Ofrece una valoración 3.3- El evaluador actualiza el registro de entrenamiento de los practicantes.
4- El practicante recibe la valoración del laboratorio.	
Prioridad:	Crítico.

Tabla 2. Especificación de casos de uso del negocio

2.1.4.5 Diagrama de actividades

“Los diagramas de actividades ayudan a describir detalladamente que es lo que pasa dentro del negocio. Ellos detallan los roles específicos que juegan las personas (trabajadores del negocio) y las actividades que realizan”. (19).

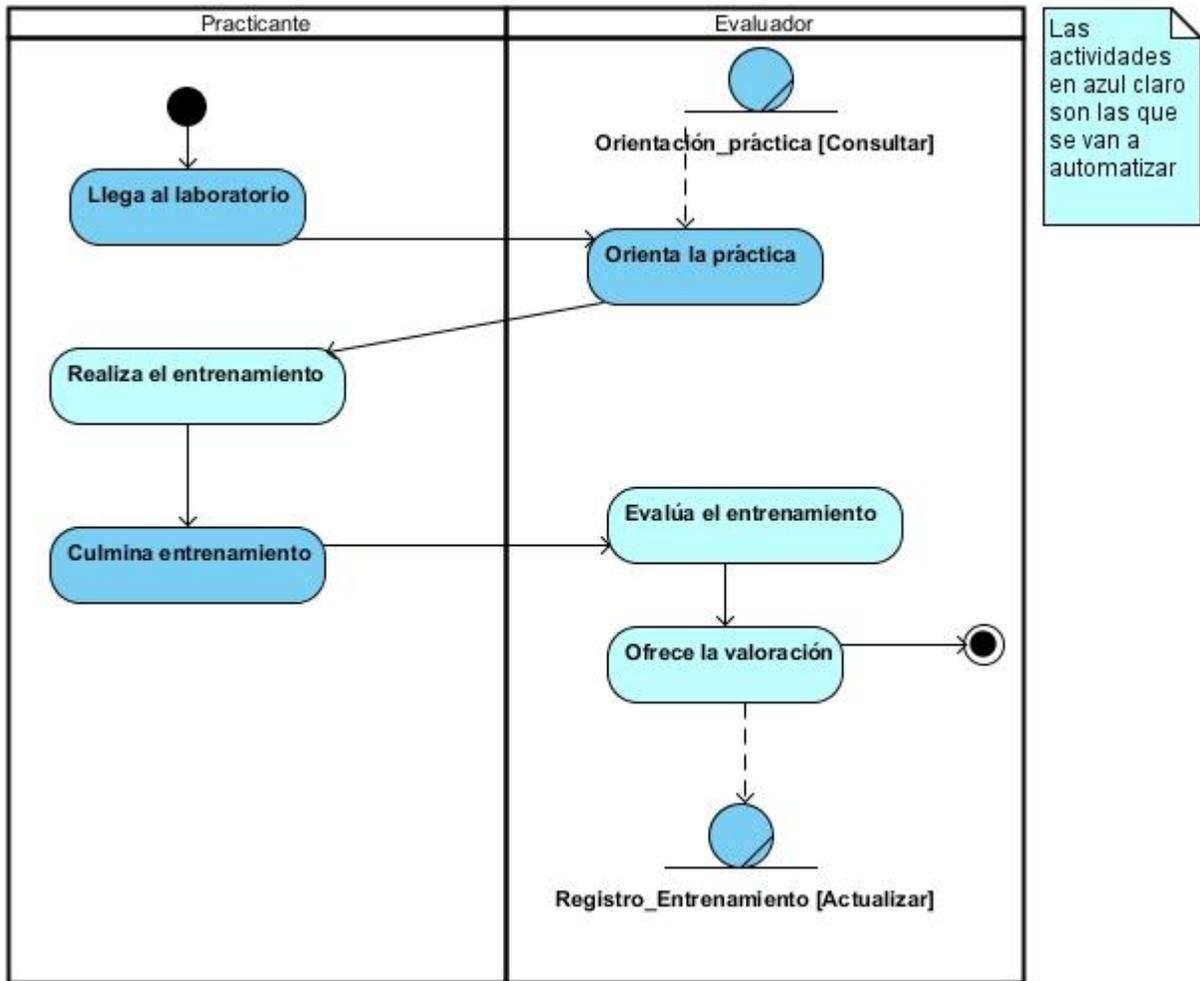


Figura 5. Diagrama de actividades

2.1.4.6 Trabajador del negocio

“Un trabajador representa a personas o sistemas (software) dentro del negocio que son las que realizan las actividades que están comprendidas dentro de un caso de uso”. (19) (Ver Figura 6)

2.1.4.7 Modelo de objetos del negocio

“El modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos.”(19)



Figura 6. Diagrama de clases del modelo de objetos del negocio

2.1.5 Especificación de requisitos

Un requisito no es más que características, cualidades, condiciones y capacidades de un sistema descritas claramente tanto para el equipo de desarrollo como para clientes y usuarios en función de satisfacer sus necesidades.

2.1.5.1 Requisitos funcionales

Los requisitos funcionales son las capacidades o funciones que el sistema debe cumplir. De acuerdo al objetivo planteado, se definen los siguientes requisitos funcionales:

RF1. Registrar usuario. El sistema permitirá al usuario registrarse como estudiante o evaluador.

RF2. Seleccionar objetos. El sistema permitirá al estudiante seleccionar los objetos que va necesitar (aros, cilindros, cuerda) para realizar los ejercicios de la práctica.

RF3. Mostrar objetos. El sistema permitirá mostrar los objetos de la práctica (aros, cilindros, cuerda) posterior a su selección.

RF4. Administrar objetos gráficos. El sistema permitirá al estudiante trabajar con los objetos gráficos (aros, cilindros, cuerda) dentro de la escena.

RF4.1 Mover objetos gráficos: El sistema permitirá al estudiante mover los objetos gráficos dentro de la escena.

RF4.2 Eliminar objeto gráfico: El sistema permitirá al estudiante eliminar el objeto gráfico que no desee en la escena.

RF5. Realizar práctica. El sistema permitirá al estudiante realizar los ejercicios orientados para la práctica.

RF5.1 El sistema permitirá tocar secciones específicas indicadas por colores.

RF5.2 El sistema permitirá seguir un camino que será dibujado en la escena del ejercicio.

RF5.3 El sistema permitirá atravesar con una pinza una sucesión de aros colocados en la escena.

RF5.4 El sistema permitirá pasar un hilo de sutura de seda por 5 argollas posicionadas a diferentes alturas.

RF5.5 El sistema permitirá seleccionar una figura para colocarla en una cuadrícula con cuatro filas y cuatro columnas.

RF5.6 El sistema permitirá colocar argollas dentro de unos cilindros dispuestos verticalmente.

RF5.7 El sistema permitirá efectuar el corte preciso de dos líneas curvas.

RF5.8 El sistema permitirá progresar por 140 cm de cuerda de una punta a la otra.

RF6. Generar reporte. El sistema permitirá al evaluador generar un reporte con su valoración de la práctica, incluyendo también las penalidades cometidas durante su realización.

RF7. Consultar reporte de la práctica. El sistema permitirá al estudiante consultar un documento con la valoración del evaluador.

2.1.5.2 Requisitos no funcionales

Los requisitos no funcionales son las propiedades o cualidades que el producto debe tener.

Seguridad

1. El sistema debe contar con protección contra acciones no autorizadas o que puedan afectar la integridad de los datos.

Usabilidad

1. Personas que posean conocimientos básicos sobre el uso de una computadora podrán utilizar el sistema.

Soporte

1. Se brindarán servicios de mantenimiento y soporte de manera automática.

Software

1. Sistema operativo Windows XP o superior así como las distintas distribuciones de Linux.

Hardware

El sistema necesitará como requisitos de hardware mínimos:

- Procesador Pentium 4 o equivalente
- 512 Mb de memoria RAM
- Hardware de video con 64 Mb de memoria

Restricciones en el diseño y la implementación

1. Debe utilizar el motor gráfico OGRE. Todo el código será en lenguaje C/C++, regido por la teoría de Programación Orientada a Objetos.

2.1.6 Modelo del sistema

El modelo del sistema permite tener un entendimiento más detallado de cómo va a estar estructurado el sistema a partir de los casos de uso (CU) identificados. Para ello se identifican los actores y se especifican los casos CU del sistema. En este epígrafe se muestra la especificación de casos de uso del sistema resultantes del levantamiento de requisitos para el laboratorio virtual.

1.1.6.1 Actores del sistema

Los actores del sistema son agentes externos, es decir, las personas u otros sistemas que interactúan con él. (Ver Tabla 3)

Actores	Justificación
Estudiante	Es el que realiza la práctica

Evaluador	Es el que ofrece una valoración de la práctica
Usuario	Es una generalización de estudiante y evaluador que inicia la funcionalidad autenticar usuario.

Tabla 3. Descripción de los actores del sistema

2.1.6.2 Diagrama de casos de uso del sistema.

Los diagramas de casos de usos del sistema muestran de manera gráfica los procesos y su relación con los actores. A continuación se muestra el diagrama de casos de uso del sistema. (Ver Figura 7).

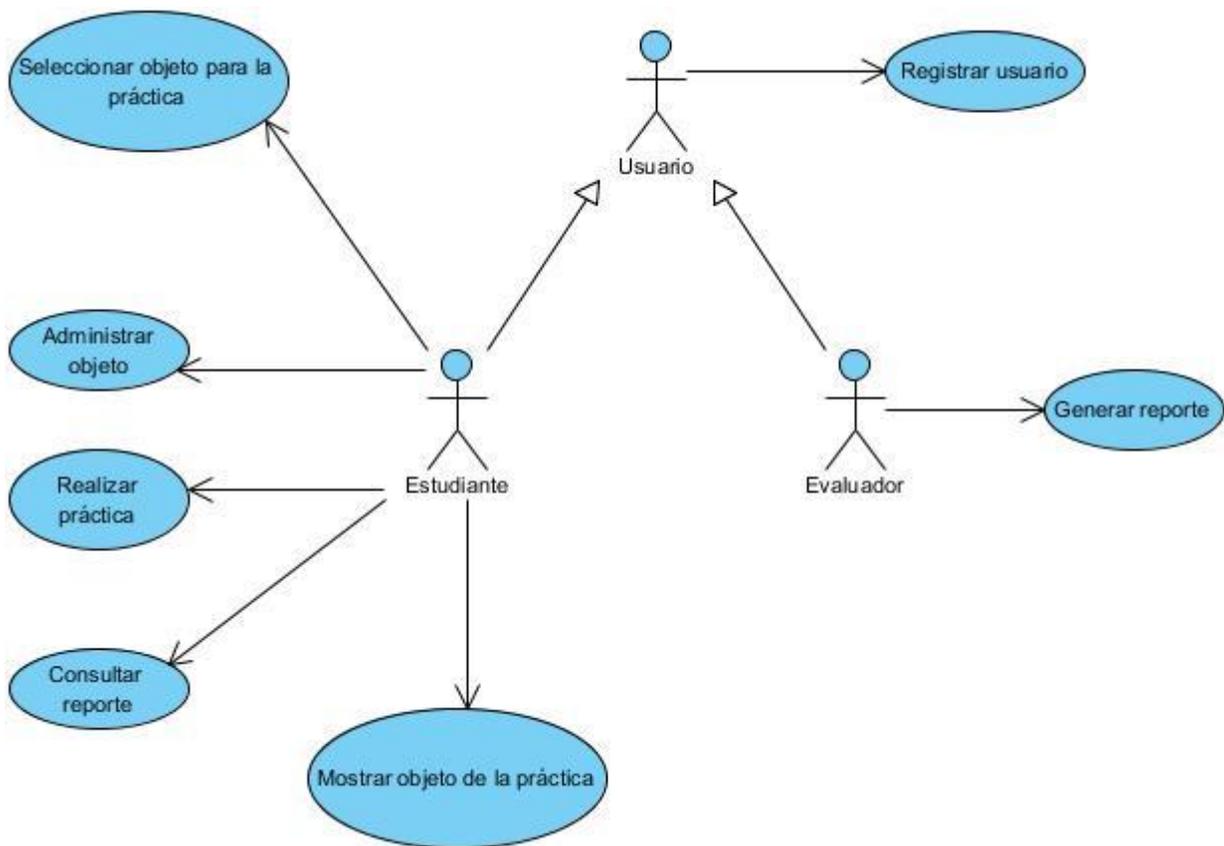


Figura 7. Diagrama de casos de uso del sistema.

2.1.6.3 Especificación de casos de uso

Objetivo	Registrarse en el sistema.	
Actores	Usuario (Inicia).	
Resumen	El caso de uso se inicia cuando el usuario desea registrarse en la aplicación y finaliza cuando accede a la misma.	
Complejidad	Alta	
Prioridad	Crítico	
Precondiciones	El usuario debe pertenecer a la especialidad cirugía.	
Postcondiciones	El usuario se registró y pudo acceder a la práctica.	
Flujo de eventos		
Flujo básico Registrar usuario		
	Actor	Sistema
1.	Accede a la realización de la práctica de laboratorio.	1.1 Muestra formulario para introducir datos.
2.	Introduce datos solicitados.	1.1 Verifica datos. 1.2 Registra usuario. 1.3 Muestra interfaz de acuerdo al tipo de usuario (estudiante o evaluador). 1.4 Termina el caso de uso.

Flujos alternos		
Nº Evento 1. Datos incorrectos		
	Actor	Sistema
		2.1a Emite un mensaje: "Datos incorrectos"
Requisitos no funcionales	Seguridad	

Tabla 4. CUS1 Registrar usuario

Objetivo	Seleccionar los objetos necesarios para realizar la práctica.	
Actores	Estudiante (Inicia).	
Resumen	El caso de uso se inicia una vez que el usuario accede a la práctica de laboratorio y termina cuando escoge los objetos a necesitar para realizar los ejercicios.	
Complejidad	Media	
Prioridad	Crítico	
Precondiciones	El usuario debe estar autenticado.	
Postcondiciones	El usuario seleccionó los objetos para realizar los ejercicios.	
Flujo de eventos		
Flujo básico Seleccionar objeto para la práctica		
	Actor	Sistema

1.	Accede a la realización de la práctica.	<p>1.1 Muestra el área de trabajo donde se va a realizar la práctica de laboratorio.</p> <p>1.2 Muestra un panel de herramientas donde se encuentran todos los objetos gráficos correspondientes a la práctica de laboratorio.</p>
2.	Selecciona el objeto que necesita para realizar la práctica.	<p>1.5 Resalta objeto seleccionado.</p> <p>1.6 Termina el caso de uso.</p>
Requisitos no funcionales		-

Tabla 5. CUS2 Seleccionar objeto de la práctica

Objetivo	Mostrar los objetos seleccionados para realizar la práctica.	
Actores	Estudiante (Inicia).	
Resumen	El caso de uso se inicia una vez que el usuario accede a la práctica de laboratorio y termina cuando se muestran los objetos seleccionados para realizar los ejercicios.	
Complejidad	Media	
Prioridad	Crítico	
Precondiciones	El usuario debe estar autenticado. Los objetos deben seleccionarse previamente.	
Postcondiciones	Todos los objetos seleccionados correspondientes a la práctica de laboratorio fueron mostrados.	
Flujo de eventos		
Flujo básico Mostrar objeto para la práctica		
	Actor	Sistema

1.	Accede a la realización de la práctica.	1.1 Muestra el área de trabajo donde se encuentran los objetos seleccionados para la práctica de laboratorio. 1.2 Termina el caso de uso.
Requisitos no funcionales	-	

Tabla 6. CUS3 Mostrar objeto de la práctica

Objetivo	Administrar los objetos gráficos	
Actores	Estudiante (Inicia)	
Resumen	El caso de uso comienza cuando el usuario accede a la práctica y termina cuando se culmina la práctica.	
Complejidad	Alta	
Prioridad	Crítico	
Precondiciones	Debe existir un objeto gráfico en la escena. El usuario debe estar autenticado.	
Postcondiciones	El estudiante movió y eliminó los objetos gráficos.	
Flujo de eventos		
Flujo básico Gestionar objetos		
	Actor	Sistema
1.	Accede a la realización de la práctica.	1.1 Muestra área de trabajo con los objetos seleccionados para utilizar en la práctica.
2.	Escoge opciones: <ul style="list-style-type: none"> Mover objeto. Ver Sección 1: 	

	<p>“Mover objeto ”</p> <ul style="list-style-type: none"> • Eliminar objeto. Ver Sección 2: “Eliminar objeto”. 	
Sección 1: “Mover objeto”		
Flujo básico Mover objeto		
	Actor	Sistema
1.	Selecciona el objeto a mover	<p>1.1 Verifica que se haya seleccionado un objeto a la vez.</p> <p>1.2 Permite mover el objeto.</p> <p>1.3 Termina el caso de uso.</p>
Flujos alternos		
Nº Evento 1. Selecciona más de un objeto		
	Actor	Sistema
1.		1.1 a Muestra mensaje “Seleccione un solo objeto a la vez”.
Sección 2: “Eliminar objeto”		
Flujo básico Eliminar objeto		
	Actor	Sistema
1.	Selecciona el objeto a eliminar	1.1 Verifica selección.
2.	Presiona botón Eliminar	<p>2.1 Elimina el objeto gráfico seleccionado.</p> <p>2.2 Termina el caso de uso.</p>
Flujos alternos		

Nº Evento 1. No selecciona ningún objeto.		
	Actor	Sistema
		1.1 a Muestra mensaje "Selecione el objeto a eliminar".
Requisitos no funcionales	-	

Tabla 7. CUS4 Administrar objeto

Objetivo	Realizar la práctica de laboratorio.	
Actores	Estudiante (Inicia)	
Resumen	El caso de uso comienza cuando el usuario accede a la práctica y termina cuando realiza los ejercicios de la práctica.	
Complejidad	Alta	
Prioridad	Crítico	
Precondiciones	El usuario debe estar autenticado. Los objetos para realizar los ejercicios deben seleccionarse previamente.	
Postcondiciones	El estudiante realizó los ejercicios de la práctica.	
Flujo de eventos		
Flujo básico Realizar práctica		
	Actor	Sistema
1.	Accede a la realización de la práctica.	1.1 Muestra un panel de herramientas con los objetos seleccionados para utilizar en la práctica.
2.	Para realizar:	

	<ul style="list-style-type: none"> • Ejercicio 1. Ver Sección 1: “Ejercicio 1”. • Ejercicio 2. Ver Sección 2: “Ejercicio 2”. • Ejercicio 3. Ver Sección 3: “Ejercicio 3” • Ejercicio 4. Ver Sección 4: “Ejercicio 4”. • Ejercicio 5. Ver Sección 5: “Ejercicio 5”. • Ejercicio 6. Ver Sección 6: “Ejercicio 6”. • Ejercicio 7. Ver Sección 7: “Ejercicio 7”. • Ejercicio 8. Ver Sección 8: “Ejercicio 8”. 	
Sección 1: “Ejercicio 1”		
Flujo básico Posicionamiento en zonas señaladas.		
	Actor	Sistema
1.	Selecciona la pinza laparoscópica.	1.1 Muestra secciones específicas indicadas por colores.
2.	Posiciona la pinza sobre las secciones indicadas.	2.1 Verifica posicionamiento. 2.2 Resalta sección.

		2.3 Termina el caso de uso.
Flujos alternos		
Nº Evento 1. Tocar una zona no señalada.		
	Actor	Sistema
		2.1 a Registra penalidad.
Sección 2: “Ejercicio 2”		
Flujo básico Recorrido de un camino.		
	Actor	Sistema
1.	Selecciona la pinza laparoscópica.	1.1 Dibuja el camino a seguir en la escena.
2.	Recorre el camino señalado.	2.1 Verifica camino recorrido. 2.2 Resalta camino. 2.3 Termina el caso de uso.
Flujos alternos		
Nº Evento 1. Salirse del camino.		
	Actor	Sistema
		2.1 a Registra la penalidad
Sección 3: “Ejercicio 3”		
Flujo básico Sucesión de aros.		
	Actor	Sistema
1.	Selecciona la pinza laparoscópica.	1.1 Dibuja en la escena una sucesión de aros.

2.	Atraviesa los aros con la pinza.	2.1 Verifica recorrido 2.2 Resalta los aros una vez terminado el recorrido 2.3 Termina el caso de uso.
Flujos alternos		
Nº Evento 1. Tocar aros durante el recorrido		
	Actor	Sistema
		2.1a Registra la penalidad.
Sección 4. "Ejercicio 4"		
Flujo básico Recorrido de argollas.		
	Actor	Sistema
1.	Selecciona la pinza laparoscópica.	1.1 Posiciona en la escena 5 argollas a distintas alturas.
2.	Agarra el hilo y atraviesa todas las argollas.	2.1 Verifica estado argollas. 2.2 Resalta argollas una vez terminado el recorrido. 2.3 Termina el caso de uso.
Flujos alternos		
Nº Evento 1. Saltarse una argolla		
	Actor	Sistema

		2.1a Registra la penalidad.
Sección 5. “Ejercicio 5”		
Flujo básico Colocación de figuras.		
	Actor	Sistema
1.	Selecciona la pinza laparoscópica.	1.1 Posiciona en la escena el tablero para colocar las figuras.
2.	Sostiene las piezas y las coloca en el tablero.	2.1 Verifica posicionamiento. 2.2 Resalta tablero cuando se colocan las piezas. 2.3 Termina el caso de uso.
Flujos alternos		
Nº Evento 1. Colocar mal una pieza		
	Actor	Sistema
		2.1a Registra la penalidad.
Sección 6. “Ejercicio 6”		
Flujo básico Coordinación motora.		
	Actor	Sistema
1.	Selecciona la pinza laparoscópica.	1.1 Muestra en la escena cilindros posicionados verticalmente.
2.	Sostiene las argollas y las introduce dentro	2.1 Verifica posicionamiento.

	de los cilindros.	2.2 Resalta los cilindros y las argollas. 2.3 Termina el caso de uso.
Flujos alternos		
Nº Evento 1. Rozar argolla con cilindro		
	Actor	Sistema
		2.1a Registra la penalidad.
Sección 7. “Ejercicio 7”		
Flujo básico Incisión de líneas.		
	Actor	Sistema
1.	Selecciona el bisturí.	1.1 Muestra en el espacio de trabajo dos líneas curvas. Cada línea curva tiene un diámetro de 4 mm.
2.	Corta las líneas señaladas.	2.1 Verifica corte. 2.2 Resalta corte. 2.3 Termina el caso de uso.
Flujos alternos		
Nº Evento 1. Desviación de la línea de corte		
	Actor	Sistema
		2.1a Registra la penalidad.

Sección 8. “Ejercicio 8”		
Flujo básico Progresión segmentaria.		
	Actor	Sistema
1.	Selecciona la pinza laparoscópica	1.1 Muestra una cuerda de 140 cm de largo, con secciones específicamente coloreadas que corresponden a intervalos marcados cada doce centímetros.
2.	Progresa por la cuerda por los intervalos marcados.	2.1 Verifica progresión. 2.2 Resalta cuerda cuando termina la progresión. 2.3 Termina el caso de uso.
Flujos alternos		
Nº Evento 1. Progresa por una zona no marcada.		
	Actor	Sistema
		2.1a Registra la penalidad.

Tabla 8. CUS5 Realizar práctica

Objetivo	Generar el reporte que consulta el estudiante una vez finalizada la práctica.
Actores	Evaluador (Inicia).
Resumen	El caso de uso inicia cuando el evaluador necesita crear el reporte con el resumen de la práctica y termina cuando lo crea.

Complejidad	Media	
Prioridad	Crítico	
Precondiciones	El usuario debe estar autenticado. El estudiante debe haber terminado la práctica.	
Postcondiciones	El evaluador generó el reporte resumen de la práctica.	
Flujo de eventos		
Flujo básico Generar reporte		
	Actor	Sistema
1.	Accede a la práctica y selecciona la opción "Crear reporte"	<p>1.1 Muestra una nueva ventana para escribir la valoración de la práctica.</p> <p>1.2 Muestra listado de penalidades cometidas durante la realización de la práctica.</p>
2.	Escribe valoración sobre la práctica. Si se cometió alguna penalidad, la selecciona del listado y presiona el botón "Aceptar".	<p>2.1 Conformar reporte y ofrece una vista previa del mismo.</p> <p>2.2 Termina el caso de uso.</p>
Requisitos no funcionales	-	

Tabla 9. CUS6 Generar reporte

Objetivo	Consultar el reporte una vez finalizada la práctica.
Actores	Estudiante (Inicia).

Resumen	El caso de uso inicia cuando el estudiante consulta el resumen de la práctica.	
Complejidad	Media	
Prioridad	Crítico	
Precondiciones	El usuario debe estar autenticado. El estudiante debe haber terminado la práctica. El reporte debe estar creado.	
Postcondiciones	El estudiante consultó el reporte resumen de la práctica.	
Flujo de eventos		
Flujo básico Gestionar penalidades		
	Actor	Sistema
1.	Accede a la práctica y selecciona la opción "Consultar reporte"	1.1 Muestra mensaje ("Seleccione ruta para almacenar reporte").
2.	Selecciona la ruta y presiona el botón "Aceptar".	2.1 Guarda reporte en la ruta seleccionada. 2.2 Muestra mensaje ("Reporte almacenado correctamente"). 2.3 Termina el caso de uso.

Tabla 10. CUS7 Consultar reporte

2.2 Análisis y diseño del sistema

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

2.2.1 Estilos arquitectónicos

“Los estilos arquitectónicos al igual que su homónimo en la arquitectura tradicional representan diferentes tendencias que de alguna manera u otra han sido reconocidas y utilizadas por los especialistas. Un estilo describe una clase de arquitectura, o piezas identificables de las arquitecturas empíricamente dadas.”(20)

2.2.1.1 Estilo basado en capas

“El estilo basado en capas funciona como una organización jerárquica tal que cada capa proporciona servicios a la capa inmediatamente superior y se sirve de las prestaciones que le brinda la inmediatamente inferior.”(21)

En la práctica, las capas suelen ser entidades complejas, compuestas de varios paquetes o subsistemas.

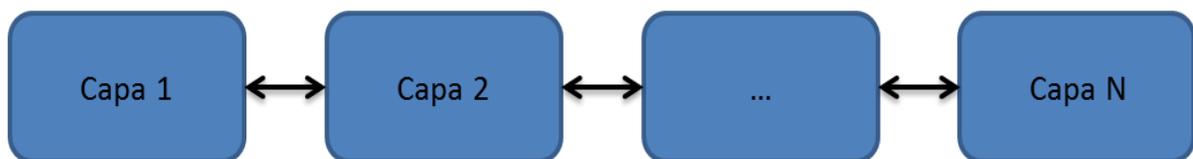


Figura 8. Estilo basado en capas

2.2.1.2 Estilo basado en componentes

“Una arquitectura basada en componentes describe una aproximación de ingeniería de software al diseño y desarrollo de un sistema. Esta arquitectura se enfoca en la descomposición del diseño en componentes funcionales o lógicos que expongan interfaces de comunicación bien definidas. Esto provee un nivel de abstracción mayor que los principios de orientación por objetos y no se enfoca en asuntos específicos de los objetos como los protocolos de comunicación y la forma como se comparte el estado.

Un componente es una unidad de composición de aplicaciones, que posee un conjunto de interfaces especificadas contractualmente y dependencias del contexto explícitas. Que ha de poder ser desarrollado, adquirido, incorporado al sistema y compuesto con otros componentes de forma independiente, en tiempo y espacio.” (22)

A partir del estudio realizado sobre los elementos y necesidades ya mencionadas se propone utilizar como estilos arquitectónicos los basados en capas y en componentes. La combinación de estos estilos permitirá amplio nivel de reutilización y facilidad de desarrollo.

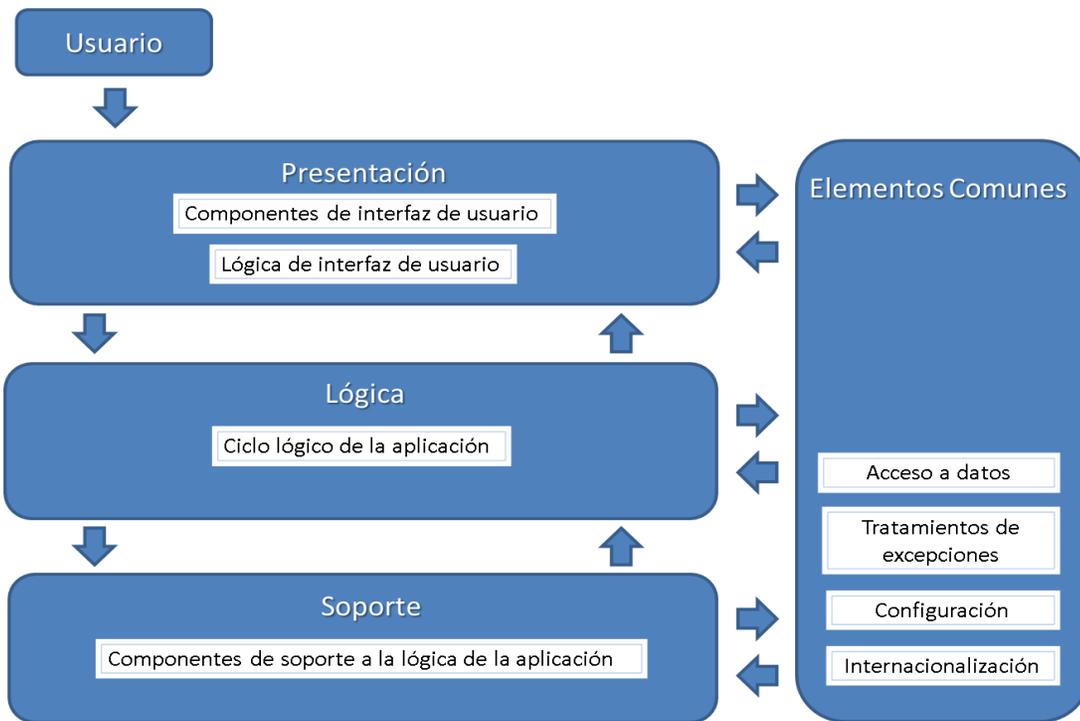


Figura 9. Arquitectura del laboratorio virtual

2.2.3 Patrones de diseño

“Los patrones de diseño son soluciones simples y elegantes a problemas específicos y comunes del diseño orientado a objetos. Son soluciones basadas en la experiencia y que se ha demostrado que funcionan.”(23).

2.2.3.1 Patrones de diseño aplicados

Las cualidades de un patrón de diseño están definidas por su encapsulamiento y abstracción pues cada patrón encapsula un problema bien definido y su solución en un dominio particular. Cada patrón una vez aplicado genera un contexto resultante.

De los patrones GoF (Gang of Four) se aplica 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.

2.2.4 Diagrama de clases del análisis.

El diagrama de clases del análisis es simplemente una vista estática de las clases que conforman el Modelo del Análisis y las asociaciones entre las mismas. Es una vista de la futura composición de clases de software. Está compuesto por clases y las relaciones entre ellas.

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

- Clases Interfaz: Modelan la interacción entre el sistema y sus actores.
- Clases Controladoras: 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”. (24).

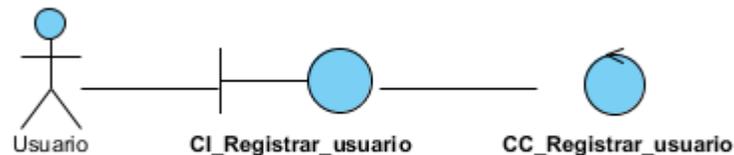


Figura 10. Diagrama de clases del análisis Registrar usuario

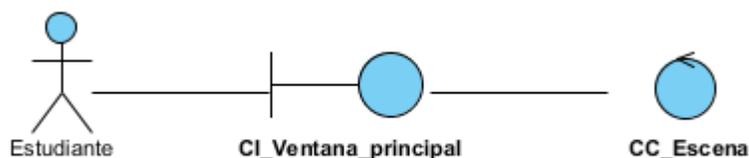


Figura 11. Diagrama de clases del análisis Seleccionar y mostrar objeto de la práctica

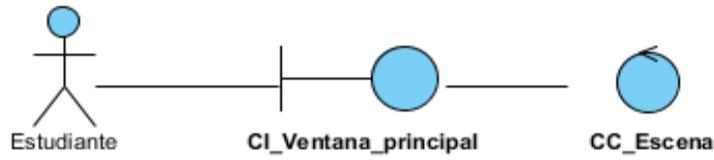


Figura 12. Diagrama de clases del análisis Administrar objeto

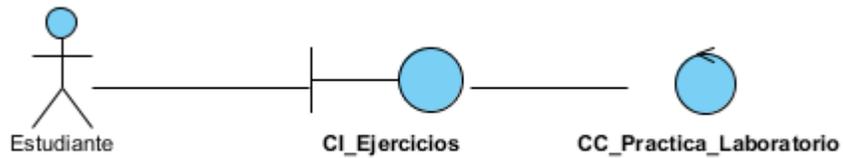


Figura 13. Diagrama de clases del análisis Realizar práctica

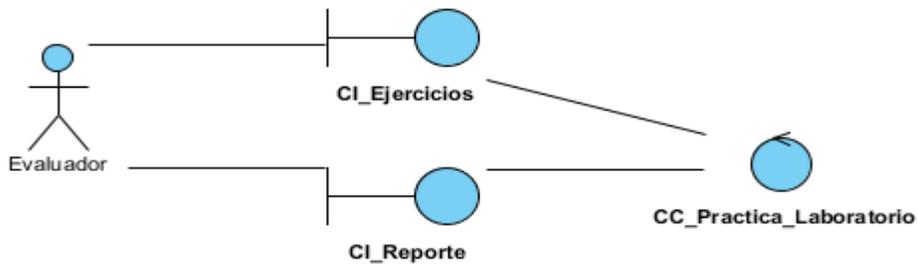


Figura 14. Diagrama de clases del análisis Generar reporte

“Los diagramas clases del análisis son un paso importante para lograr que los desarrolladores tengan un entendimiento profundo de cada uno de los procesos del sistema, es importante destacar que son la base de la arquitectura que se le vaya a aplicar al software, sin tener que depender de los lenguajes de programación.” (24)

2.2.5 Diagrama de interacción

“Un diagrama de interacción muestra una interacción, que consiste en un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos. Representa la forma en cómo un cliente (actor) u objetos (clases) se comunican entre sí en petición a un evento.” (24)

2.2.5.1 Diagramas de colaboración

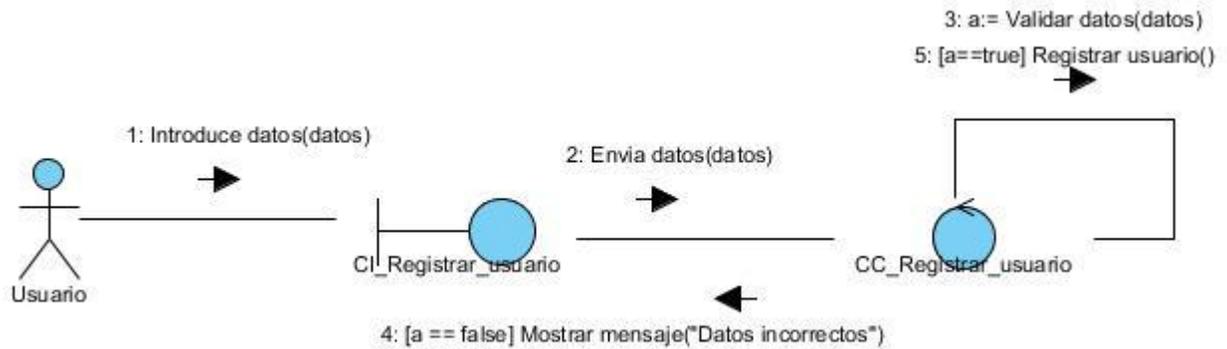


Figura 15. Diagrama de colaboración Registrar usuario

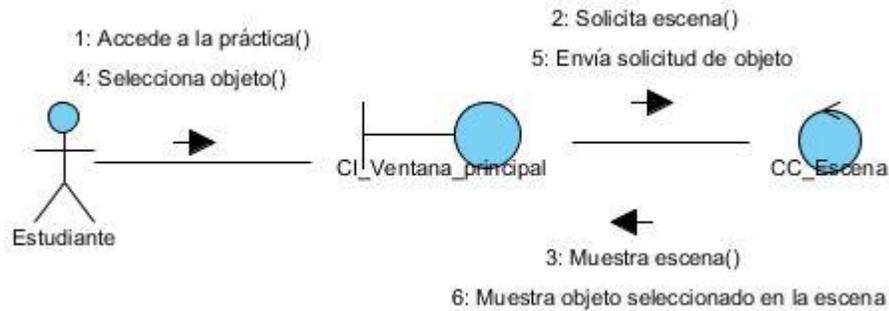


Figura 16. Diagrama de colaboración Seleccionar y mostrar objeto de la práctica

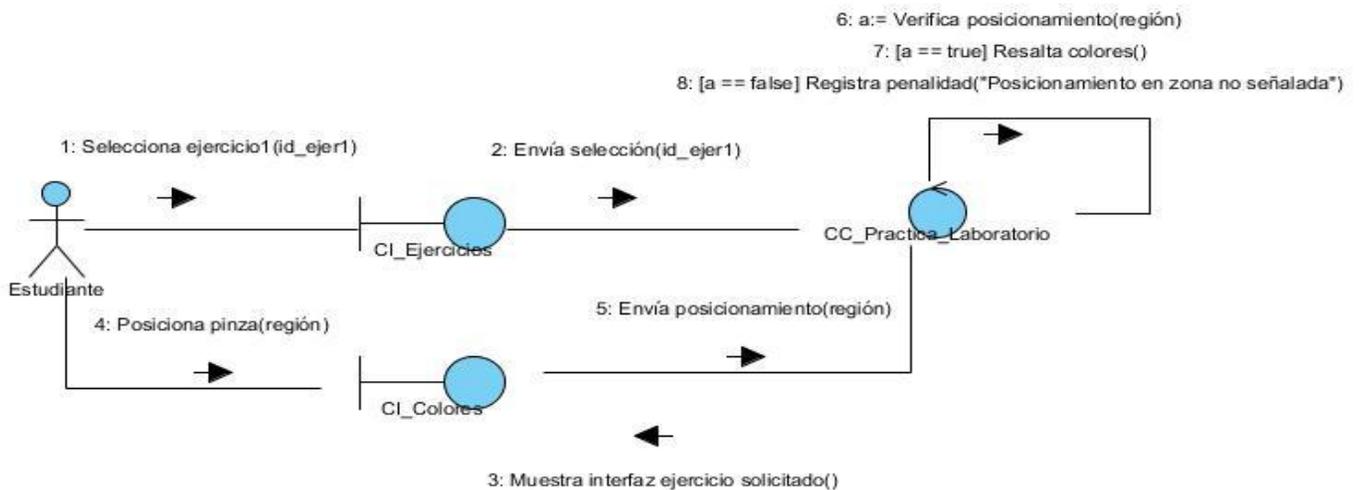


Figura 17. Diagrama de colaboración Realizar práctica sección Ejercicio 1

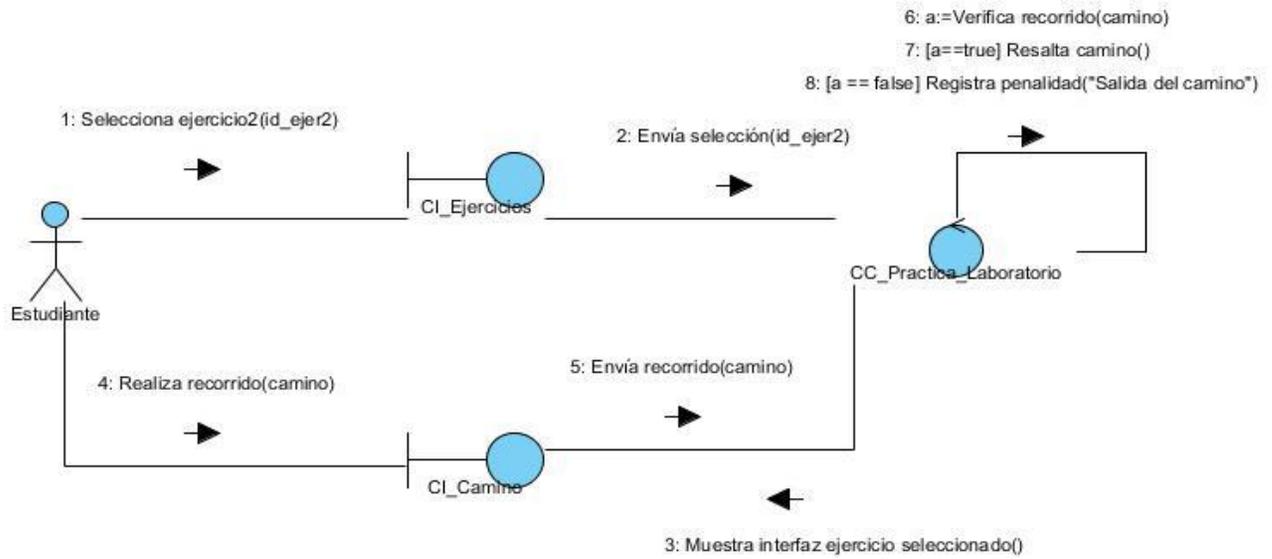


Figura 18. Diagrama de colaboración Realizar práctica sección Ejercicio 2

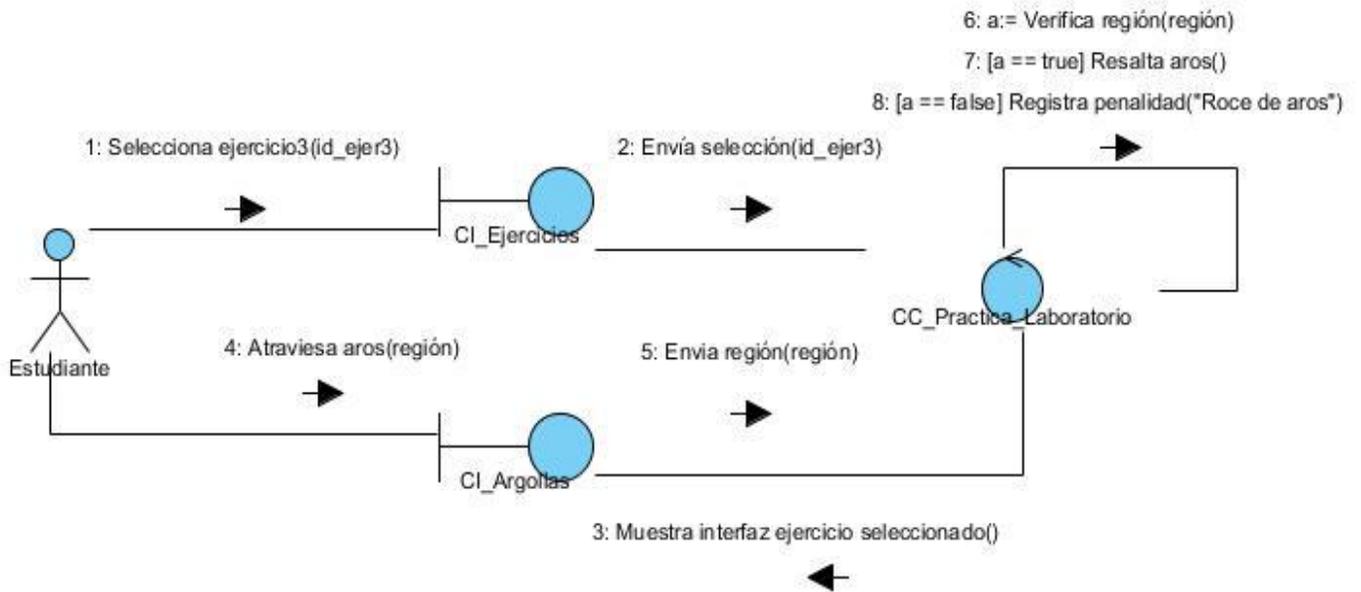


Figura 19. Diagrama de colaboración Realizar práctica sección Ejercicio 3

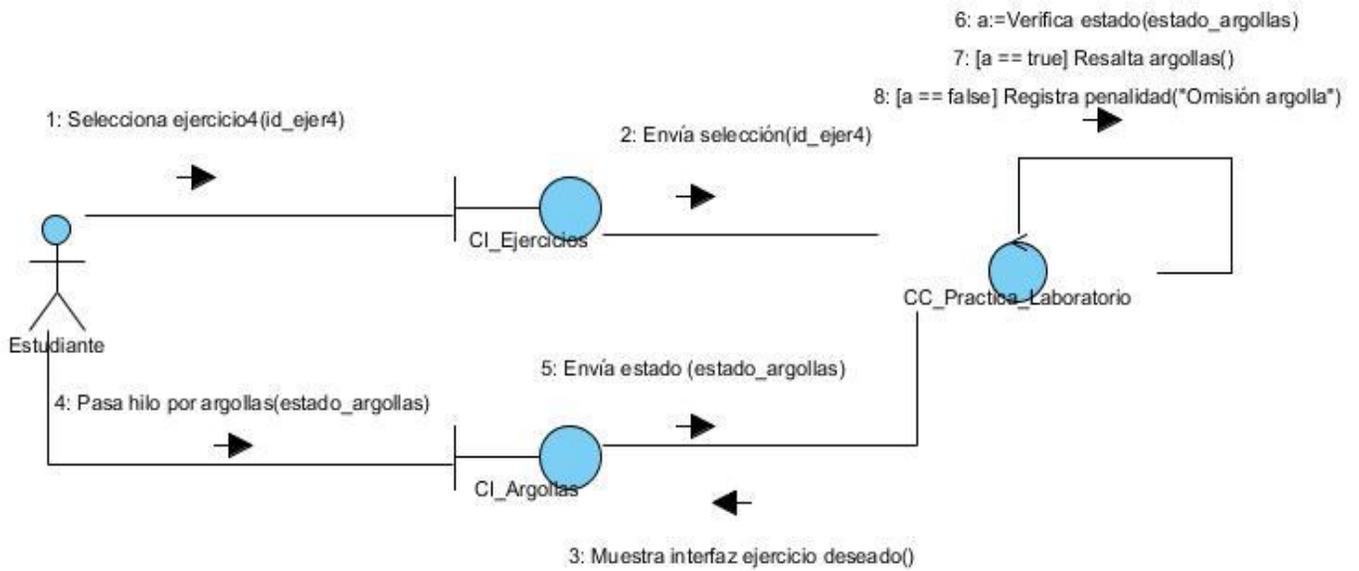


Figura 20. Diagrama de colaboración Realizar práctica sección Ejercicio 4

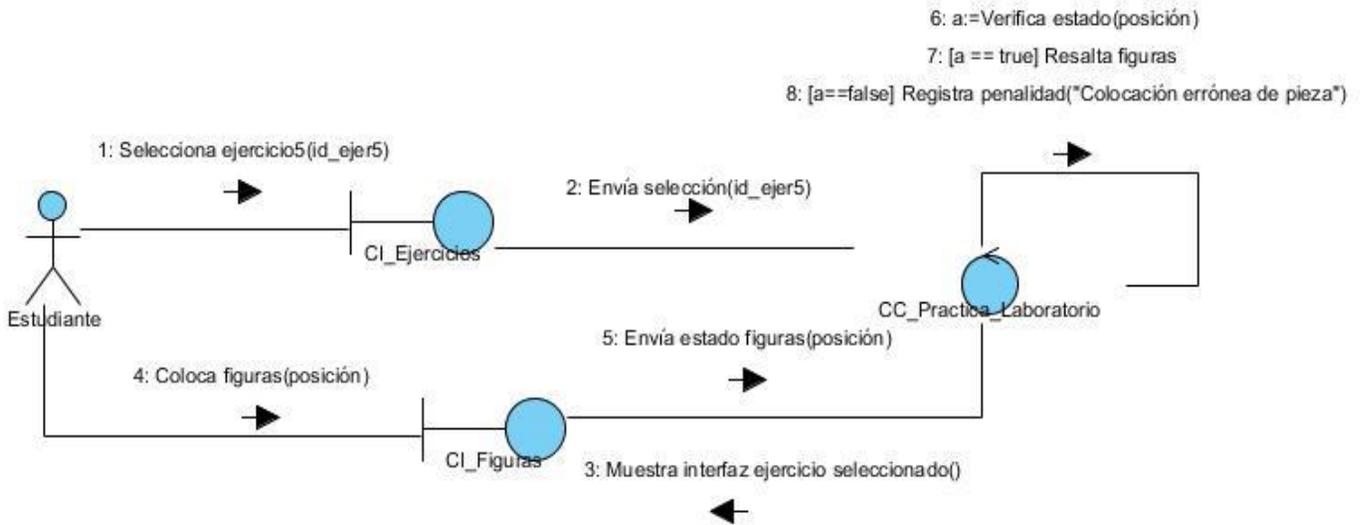


Figura 21. Diagrama de colaboración Realizar práctica sección Ejercicio 5

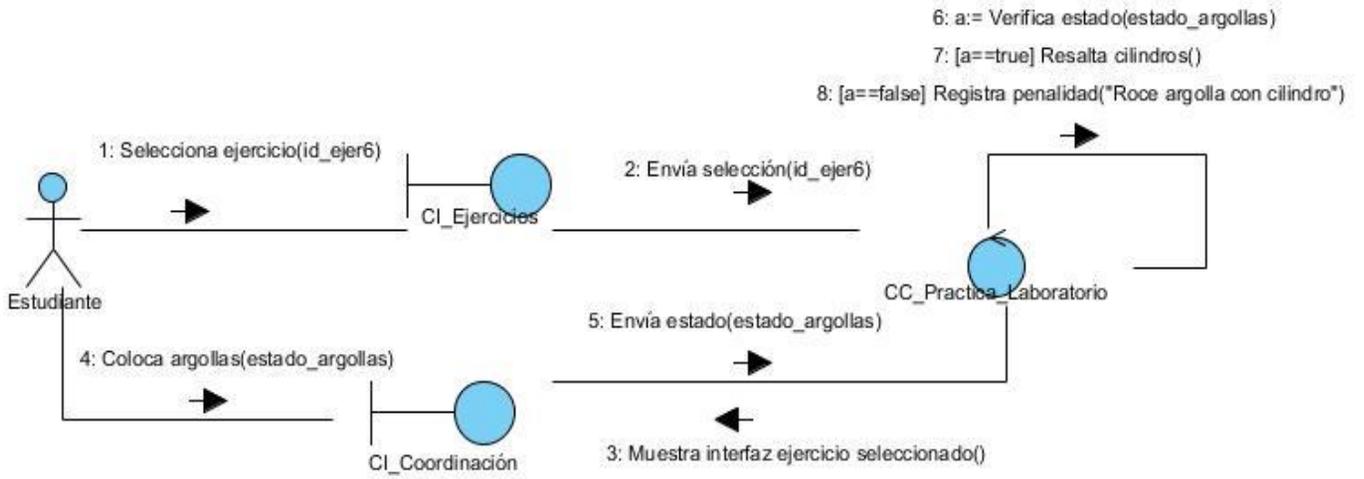


Figura 22. Diagrama de colaboración Realizar práctica sección Ejercicio 6

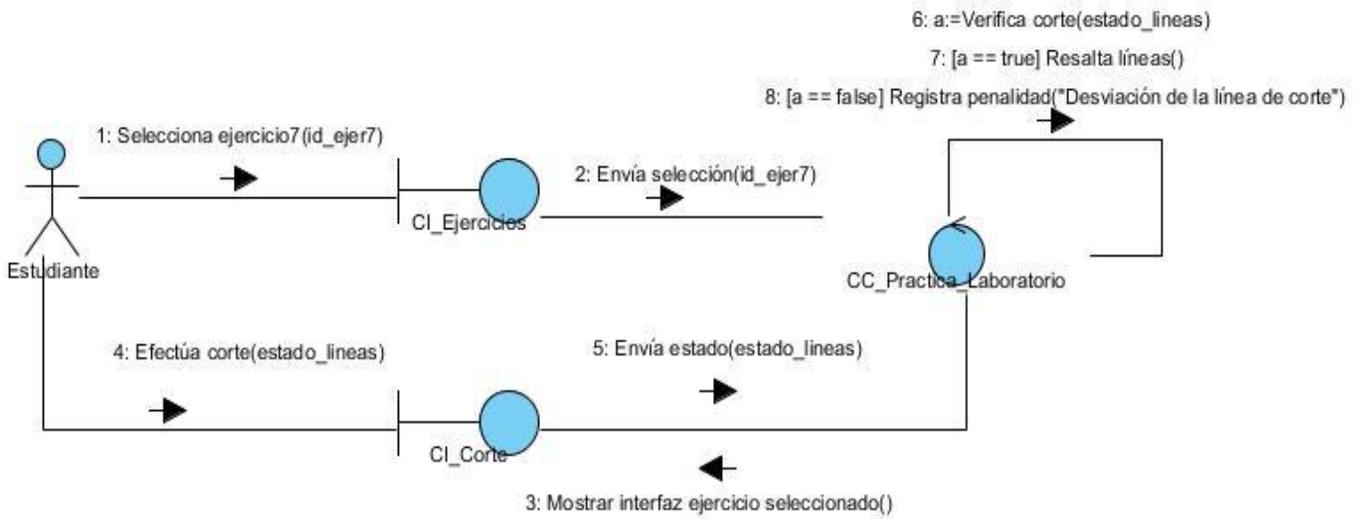


Figura 23. Diagrama de colaboración Realizar práctica sección Ejercicio 7

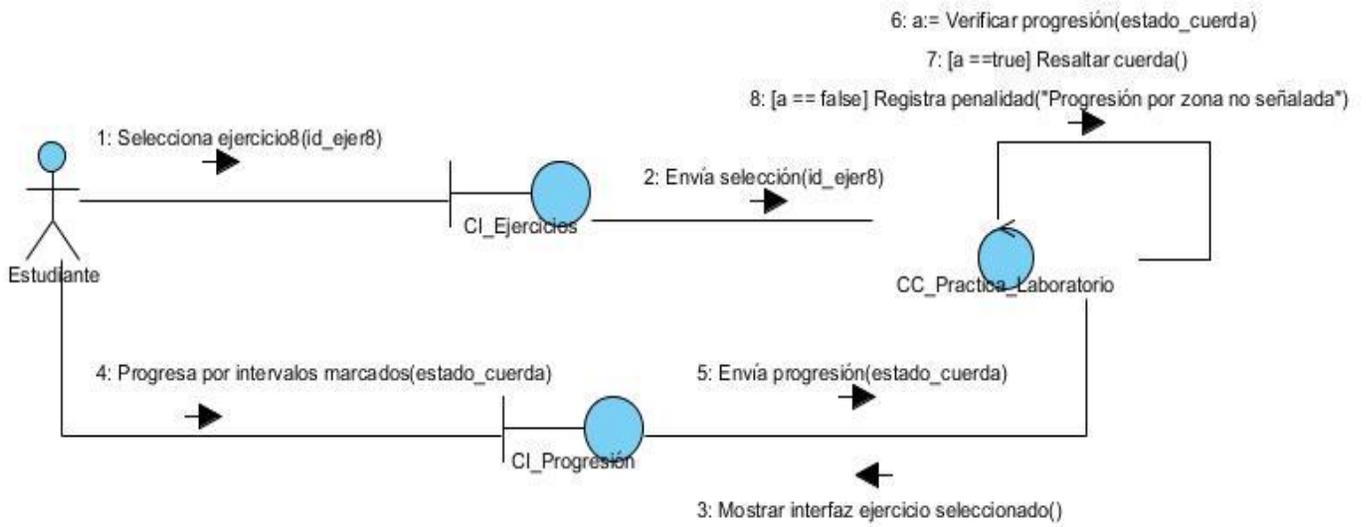


Figura 24. Diagrama de colaboración Realizar práctica sección Ejercicio 8

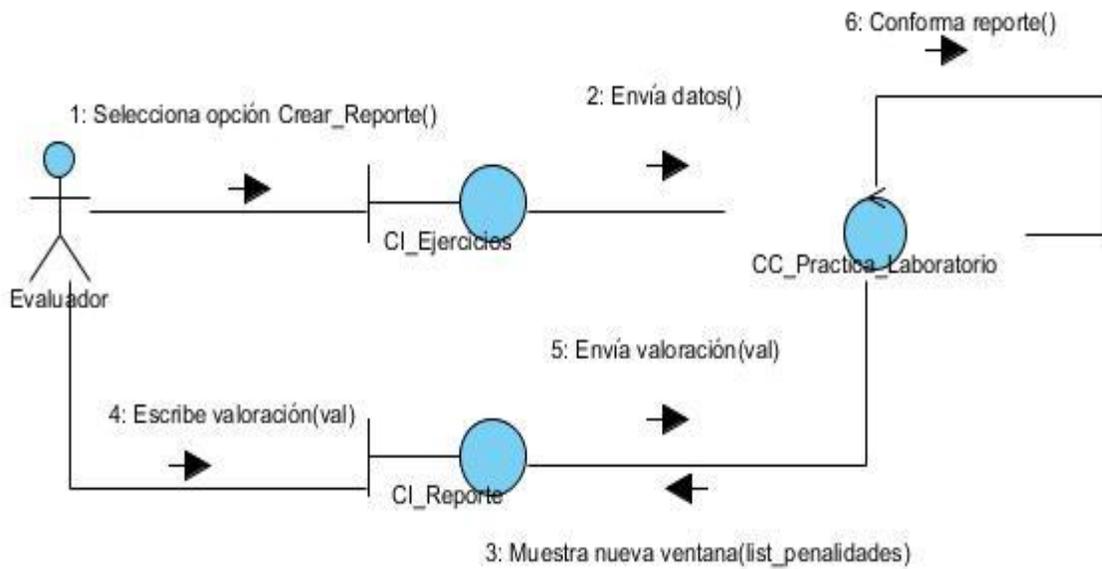


Figura 25. Diagrama de colaboración Generar reporte

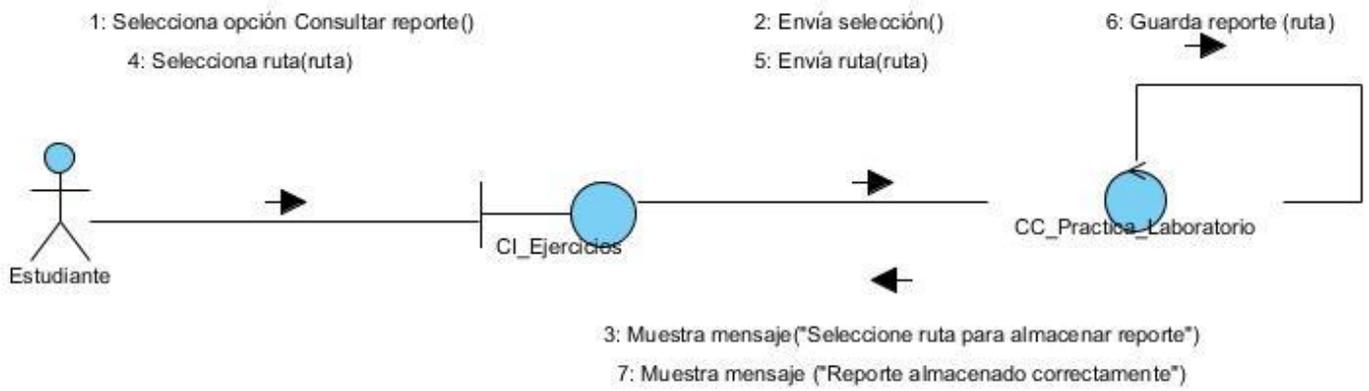


Figura 26. Diagrama de colaboración Consultar reporte

2.2.6 Diagrama de clases del diseño

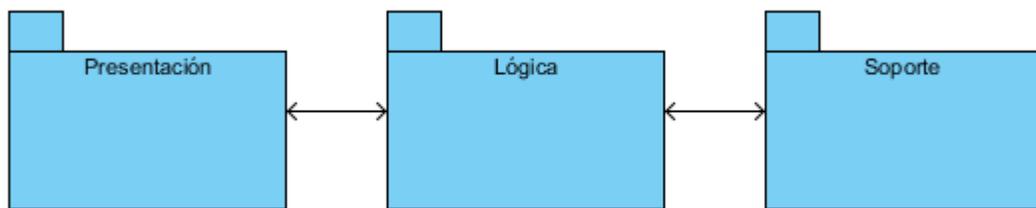


Figura 27. Paquetes de las clases del diseño

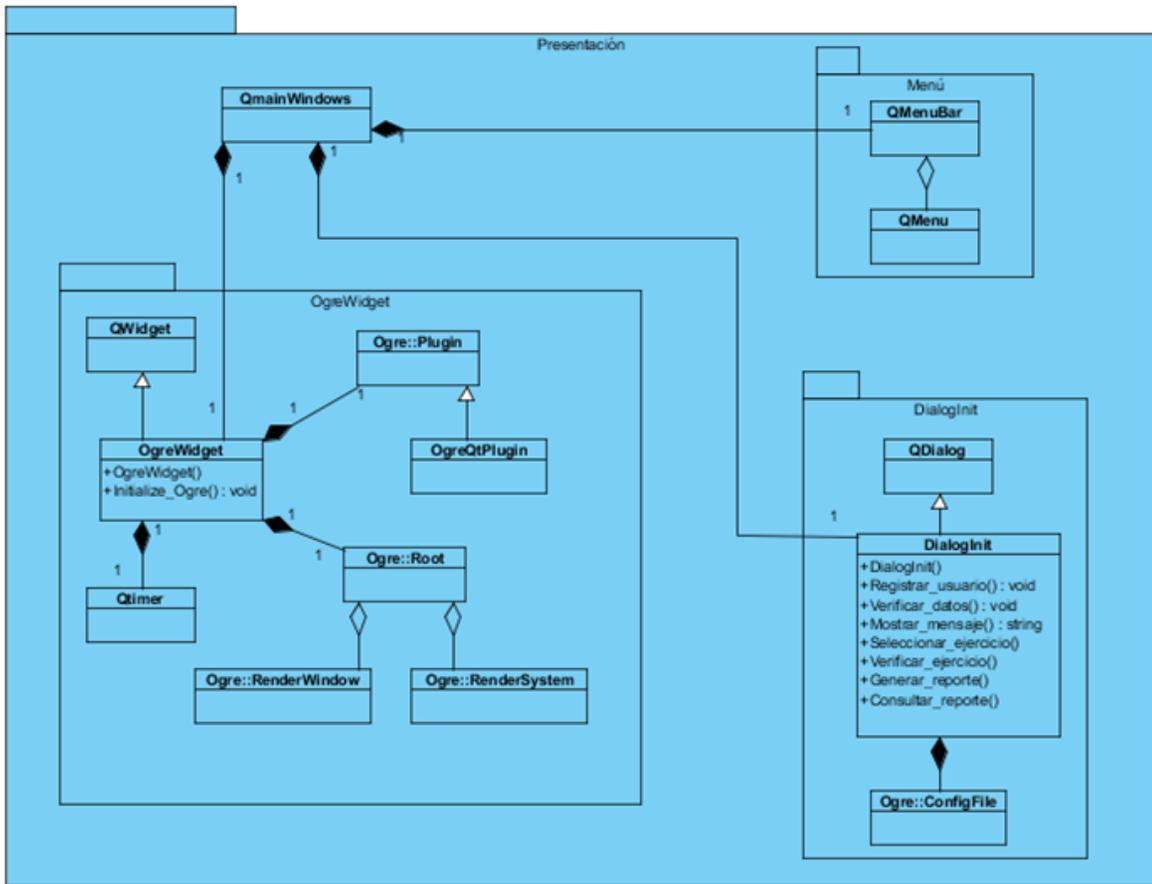


Figura 28. Capa de presentación

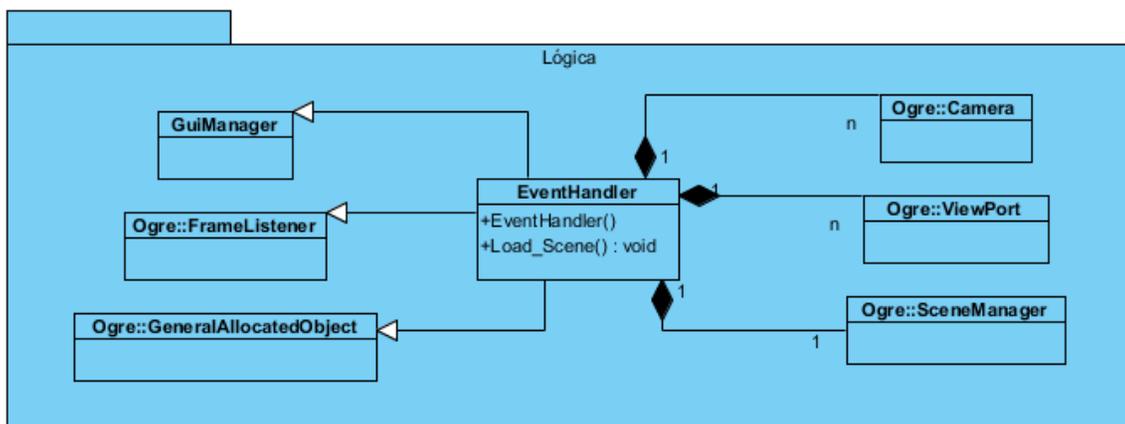


Figura 29. Capa Lógica

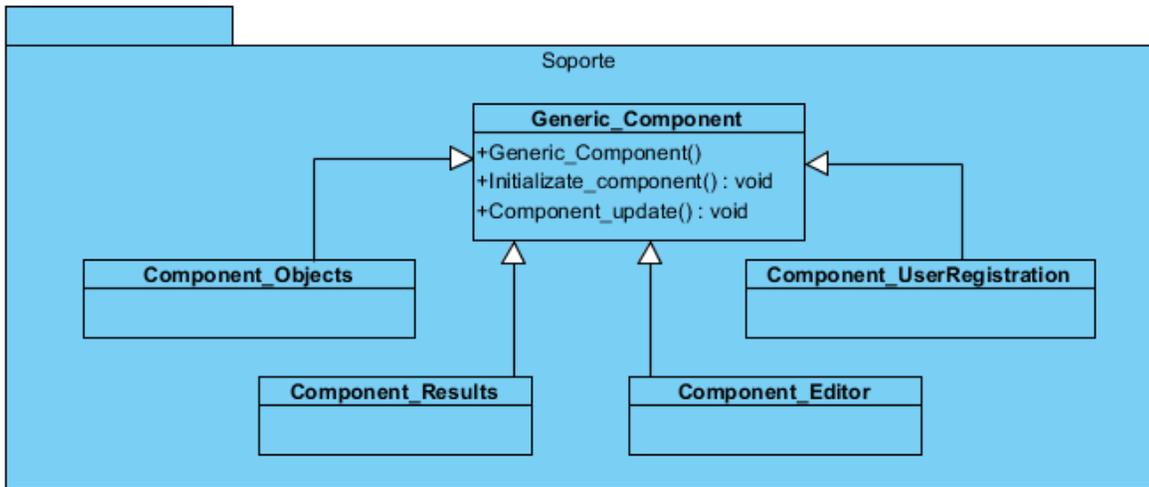


Figura 30. Capa Soporte

2.2.6.1 Diagrama de secuencia del diseño

El diagrama de secuencia es un diagrama de interacción que destaca la ordenación temporal de los mensajes. A continuación los diagramas de secuencias propuestos para la implementación del laboratorio:

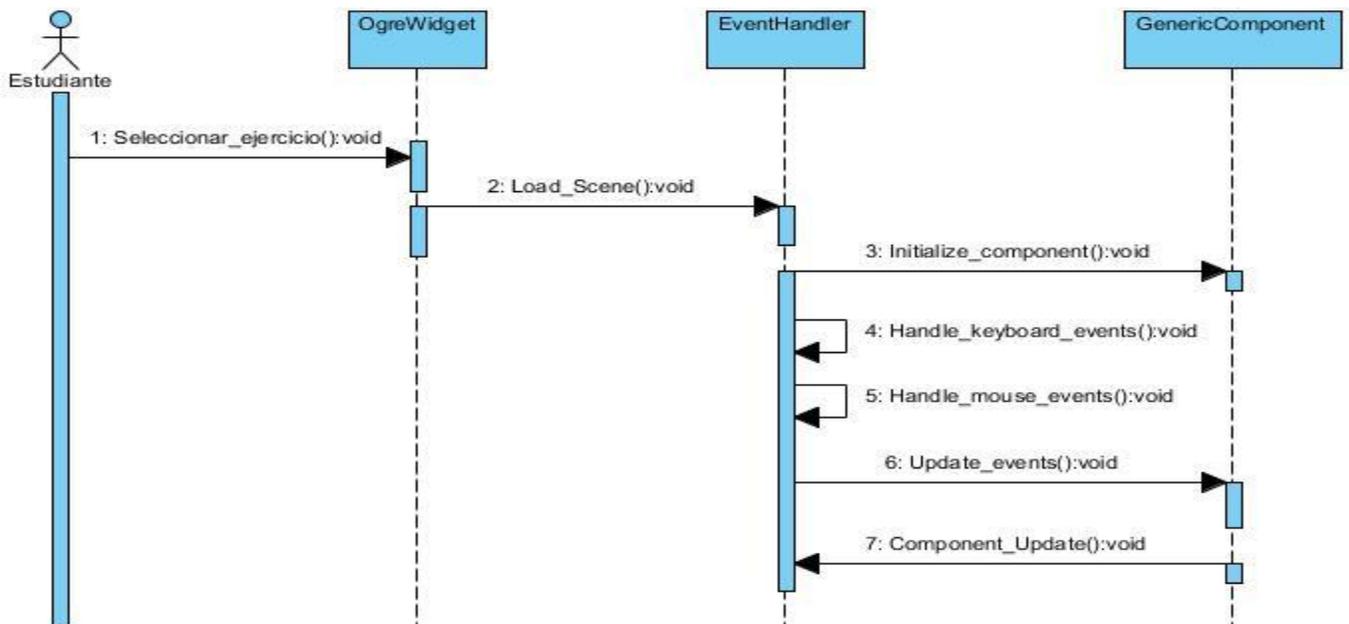


Figura 31. Diagrama de secuencia Realizar práctica

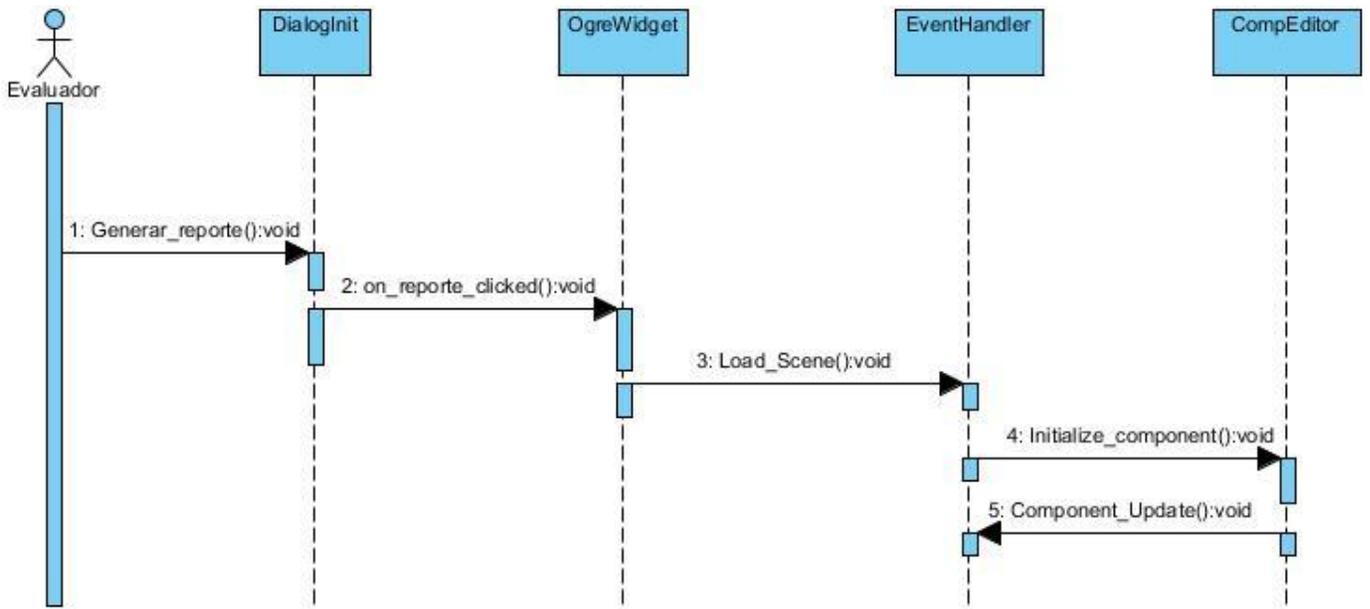


Figura 32. Diagrama de secuencia Generar reporte

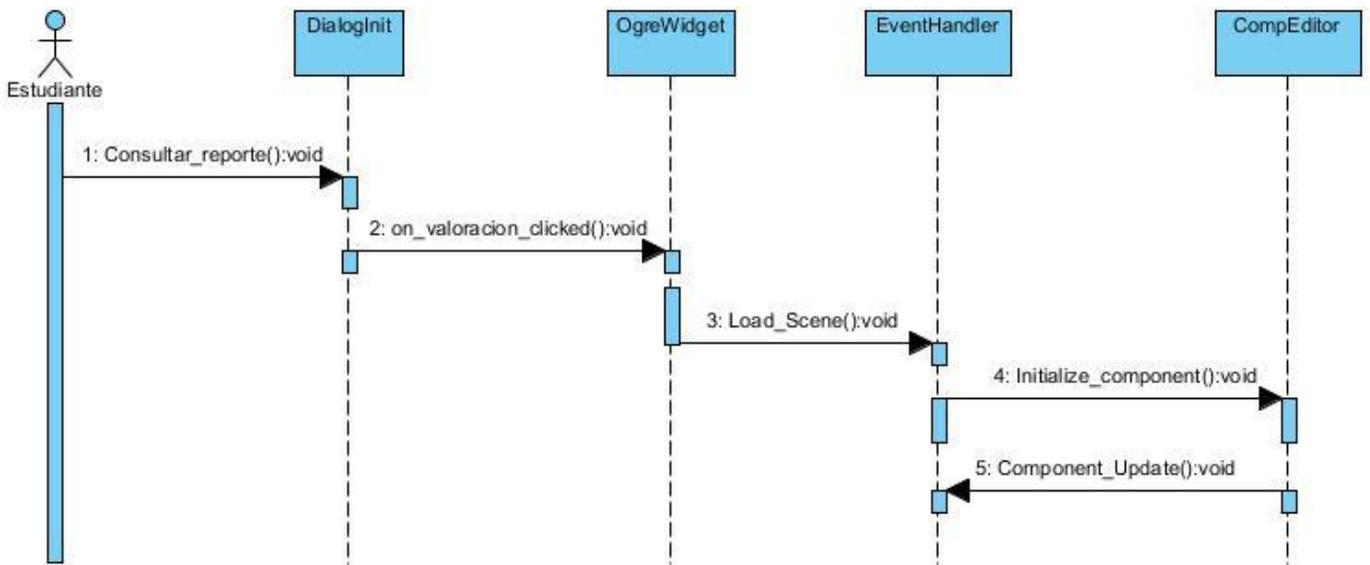


Figura 33. Diagrama de secuencia Consultar reporte

2.2.7 Diagrama de Despliegue

Los diagramas de despliegue describen la arquitectura física del sistema durante la ejecución y la topología del sistema: la estructura de los elementos de hardware y software que ejecuta cada uno de ellos. El diagrama de despliegue está constituido por nodos físicos en los cuales se ejecutan los componentes.

La práctica de laboratorio necesita para su ejecución satisfactoria una PC Cliente donde se encontrará la práctica de laboratorio instalada.

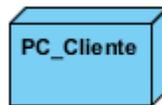


Figura 34. Diagrama de despliegue

Consideraciones parciales

- Se definieron debidamente los requisitos funcionales y no funcionales.
- Se aplicaron los estilos de arquitectura y patrones de diseño necesarios para lograr un diseño eficiente.
- 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, así como el diagrama de clases del diseño y el diagrama de despliegue.

CAPÍTULO 3. VALIDACIÓN DE LA PROPUESTA DE DISEÑO

Introducción

En este capítulo se valida la propuesta de diseño mediante un prototipo de características seleccionadas. Se describen dichas características y su proceso de desarrollo.

3.1 Validación con prototipos

Una de las formas más habituales y convenientes de analizar el sistema consiste en construir un prototipo, pues es una versión operativa preliminar del sistema para fines de demostración y evaluación. “Una ventaja fundamental que presenta la construcción de prototipos desde el punto de vista de la validación radica en que estos modelos, una vez construidos, pueden ser evaluados directamente por los usuarios o expertos en el dominio para validar sobre ellos el análisis y el diseño del sistema”. (25).

3.1.1 Propósitos del prototipo

“En la fase de Análisis, su principal propósito es obtener y validar los requisitos 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. En la fase de Diseño, su propósito, basándose en los requisitos 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. Además para ayudar a evaluar muchos aspectos de la implementación seleccionada.” (25).

3.1.2 Clases de prototipo

Existen varias clases de prototipo, a continuación se ofrecen sus principales características:

- **Prototipo Corregido:** Es la construcción de un sistema que funciona pero se corrige simultáneamente. En la ingeniería a este enfoque se le llama elaboración de una tabla experimental.
- **Prototipo No Funcional:** Modelo no funcional creado para probar ciertos aspectos del diseño.

- **Primer Prototipo de una serie:** Creación de un primer modelo a escala completa de un sistema llamado piloto.
- **Prototipo de características seleccionadas:** Creación de un modelo funcional que incluya algunas, pero no todas las características del sistema final.

La clase de prototipo que se utiliza para validar la propuesta de diseño es el prototipo de características seleccionadas. Mediante este modelo se pueden validar aspectos del análisis y el diseño pues es una representación limitada de las características del laboratorio virtual.

3.1.3 Prototipo de características seleccionadas

En el presente epígrafe se presenta imágenes del prototipo construido para el laboratorio virtual de teniendo en cuenta las funcionalidades definidas para el sistema.

En la Figura 35 la ventana que se muestra está compuesta por un formulario y la lista de los ejercicios propuestos para el laboratorio.

The image shows a window titled "Ejemplo" with a standard Windows-style title bar. Inside, there are two main sections. The first section, "Datos del usuario", contains two text input fields: "Nombres y Apellidos" with the text "Sonia Franco" and "Grupo" with the text "5501". The second section, "Ejercicios a realizar en la práctica.", contains a list of eight radio buttons labeled "Ejercicio 1" through "Ejercicio 8". The "Ejercicio 1" radio button is selected. At the bottom right of the window are two buttons: "Aceptar" and "Cancelar".

Figura 35. RF1 Registrar usuario

En la Figura 36 se observa un panel de herramientas donde están listados los objetos necesarios para la realización de los ejercicios.

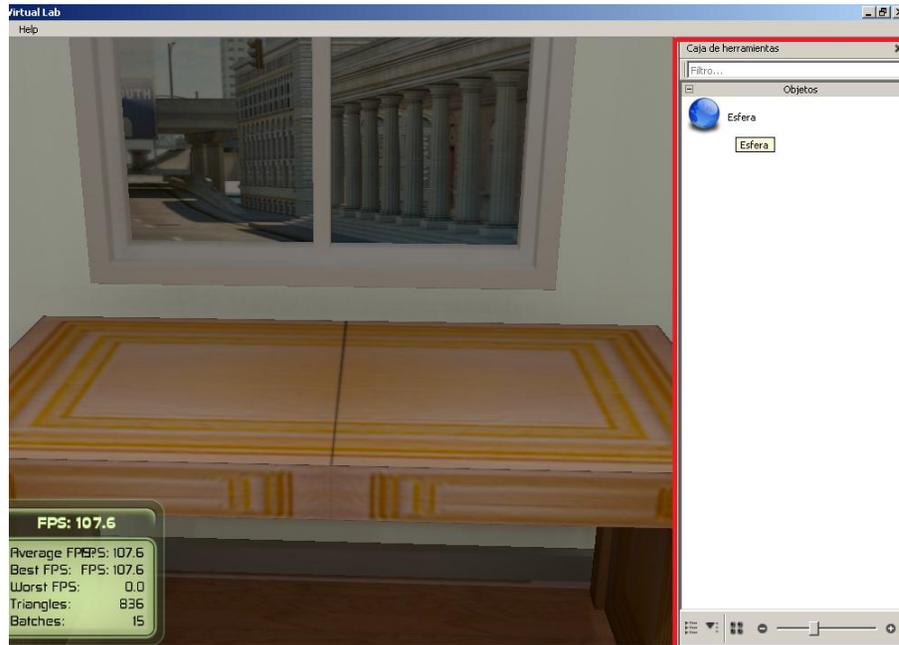


Figura 36. RF2 Seleccionar objetos

En la Figura 37 se muestran en la escena los objetos seleccionados del panel de herramientas.



Figura 37. RF3 Mostrar objetos

En la Figura 49 el objeto seleccionado de la escena se elimina.

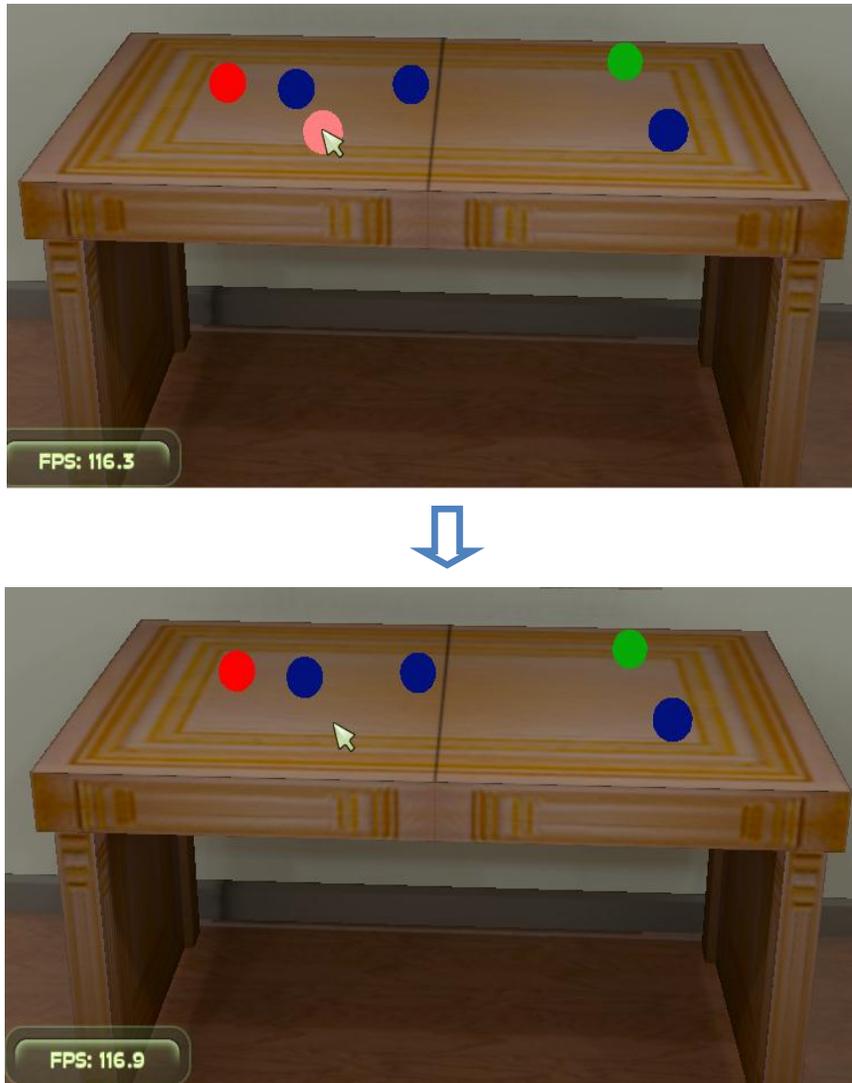


Figura 38. RF4.2 Eliminar objetos

Consideraciones parciales

- Se analizaron las distintas clases de prototipos.
- La elaboración del prototipo de características seleccionadas permitió validar la propuesta de diseño del laboratorio virtual de entrenamiento laparoscópico.

CONCLUSIONES GENERALES

Con la culminación de este trabajo se obtuvo una propuesta de diseño de un laboratorio virtual de entrenamiento laparoscópico para lo cual:

- Se realizó un estudio del estado de arte acerca del desarrollo de los laboratorios virtuales y de la cirugía de mínimo acceso.
- Se utilizaron herramientas libres y multiplataforma.
- Se definieron debidamente los requisitos funcionales y no funcionales.
- Se obtuvieron los artefactos necesarios, según la metodología de desarrollo de software seleccionada.
- Se definieron los estilos de arquitectura y patrones de diseño a aplicar para desarrollar la propuesta de solución.
- Se implementó un prototipo de características seleccionadas que permitió validar la propuesta de diseño del laboratorio virtual de entrenamiento laparoscópico.

RECOMENDACIONES

- Incrementar los ejercicios de la práctica para ampliar las posibilidades de trabajo con la herramienta.
- Implementar el laboratorio.

REFERENCIAS BIBLIOGRÁFICAS

1. P. JAMES VARY, 2000. Informe de la reunión de expertos sobre laboratorios virtuales. [Consultado 4 diciembre 2011] [Disponible en: <http://unesdoc.unesco.org/images/0011/001191/119102s.pdf>].
2. GUSTAVO ADOLFO MENESES BENAVIDES, CESAR EUGENIO ORDOSGOITIA MORALES, 2009. Laboratorio Virtual Basado en la Metodología de aprendizaje basado en problemas, ABP. [Consultado 4 diciembre 2011]. [Disponible en: http://www.acofi.edu.co/revista/revista7/2009_I_47.pdf].
3. ARIS. CARRIÓ, 2005. Diseño de recursos multimedia de biología para un aprendizaje autónomo basado en problemas. [Consultado 6 enero 2012. [Disponible en: http://ddd.uab.cat/pub/edlc/edlc_a2005nEXTRA/edlc_a2005nEXTRAp428disrec.pdf].
4. LORANDI MEDINA A. P, HERMIDA SABA G, HERNÁNDEZ SILVA J, 2010. Los Laboratorios Virtuales y Laboratorios Remotos en la Enseñanza de la Ingeniería. Revista Internacional de Educación en Ingeniería, Vol. 4, Páginas 24-29. [Consultado 9 enero 2012]. [Disponible en <http://academiajournals.com/downloads/LorandiLabsEd11.pdf>].
5. GUEVARA NUÑEZ, ORLANDO. Cirugía de mínimo acceso: ventaja, bondad y beneficio. Periódico Sierra Maestra. [Consultado 10 enero 2012]. [Disponible en <http://www.sierramaestra.cu>]
6. MEIER, U., MONSERRAT C., ALCANIZ M. Simulación Quirúrgica. Medical Image Computing Laboratory (MedICLab), Universidad Politécnica de Valencia, Valencia, Spain. [Consultado 10 enero 2012] [Disponible en <http://www.dsic.upv.es/~cmonserr/Articulos/Simul2003.pdf>.]
7. MONSERRAT C, LÓPEZ O, ALCANIZ M. Estado del Arte en simulación quirúrgica. Informática y Salud. 2004 [Consultado 11 enero 2012] [Disponible en http://scielo.isciii.es/scielo.php?pid=S0210-48062006000500006&script=sci_arttext].
8. PÁGINA OFICIAL SIMULAB [Consultado 11 enero 2012] [Disponible en: <http://www.simulab.com/product/packaged-savings/laptrainer-simuvision-package>]

9. ROQUE GONZÁLEZ, ROSALBA, RUIZ TORRES, JULIÁN, MARTÍNEZ ALONSO, MIGUEL A, TORRES PEÑA, RAFAEL, FERNÁNDEZ ZULUETA, ARNULFO, BARRIOS OSUNA, IRENE. Experiencia del Centro Nacional de Cirugía de Mínimo Acceso en educación de posgrado en cirugía laparoscópica .Rev Cubana Cir v.50 n.2 Ciudad de la Habana abr.-jun. 2011. [Consultado 16 enero 2012] [Disponible en http://www.imbiomed.com.mx/1/1/articulos.php?id_revista=57]
10. ROBERTH G. FIGUEROA, CAMILO J. SOLÍS, ARMANDO A. CABRERA, 2008. Metodologías tradicionales vs. Metodologías ágiles. [Consultado 20 enero 2012. [Disponible en <http://adonisnet.wordpress.com/2008/06/18/metodologias-tradicionales-vs-metodologias-agiles/>].
11. MANIFIESTO ÁGIL. [Consultado 20 enero 2012]. [Disponible en <http://www.agile-spain.com/tema/metodologias/>]
12. ¿QUÉ ES UML? [Consultado 20 enero 2012] [Disponible en http://www.magma.com.ni/~jorge/upoli_uml/refs/Que_es_UML.doc]
13. RATIONAL ROSE ENTERPRISE EDITION [Consultado 20 enero 2012] [Disponible en http://www.ecured.cu/index.php/Rational_Rose_Enterprise_Edition]
14. VISUAL PARADIGM. [Consultado 20 enero 2012][Disponible en http://www.ecured.cu/index.php/Visual_Paradigm]
15. PÁGINA OFICIAL DE C++. [Consultado 21 enero 2012] [Disponible en <http://www.cplusplus.com/info/description/>]
16. PÁGINA OFICIAL DE OGRE. [Consultada 21 enero 2012] [Disponible en <http://www.ogre3d.org>]
17. ECLIPSE PLATFORM TECHNICAL OVERVIEW, 2003. [Consultado: 21 enero 2012]. [Disponible en: <http://eclipse.org/whitepapers/eclipse-overview.pdf>].
18. SITIO OFICIAL QT CREATOR. [Consultado: 21 enero 2012]. [Disponible en: <http://qt.nokia.com/products/developer-tools/>]
19. FASE_DE_INICIO._DISCIPLINA_DE_MODELAMIENTO_DEL_NEGOCIO. Universidad de Ciencias Informáticas, Cuba. [Consultado 23 enero 2012]. [Disponible en: <http://eva.uci.cu/>].

20. KAZMAN, RICK, 2001. Software architecture. Handbook of Software Engineering and Knowledge Engineering [Consultado 25 enero 2012] [Disponible en: <http://www.informit.com/store/product.aspx?isbn=020170482X>].
21. GARLAN, DAVID, SHAW, MARY. An introduction to software architecture. Carnegie Mellon University. School of Computer Science. [Consultado 26 enero 2012] [Disponible en: http://sunner.cn/courses/SA/intro_softarch.pdf].
22. SZYPERSKI, CLEMENS ALDEN, 2002. Component software: beyond object-oriented programming. [Consultado 29 enero 2012] [Disponible en: <http://research.microsoft.com/en-us/um/people/cszypers/Books/component-software.htm>].
23. JOAQUIN GRACIA, 2005. Patrones de diseño: Diseño de Software Orientado a Objetos. [Consultado: 12 febrero 2012]. [Disponible en: <http://www.ingenierosoftware.com/analisisydiseño/patrones-diseño.php>].
24. INTRODUCCIÓN A LA DISCIPLINA DE ANÁLISIS Y DISEÑO. Universidad de Ciencias Informáticas, Cuba. [Consultado: 15 febrero 2012]. [Disponible en: <http://eva.uci.cu/>].
25. KENNETH E. KENDALL, JULIE E. KENDALL, ANTONIO NÚÑEZ, 2005. Análisis y diseño de sistemas. [Consultado 15 febrero 2012] [Disponible en: <http://es.scribd.com/doc/16126833/Analisis-y-diseño-de-sistemas-Kendall-Kendall>]