

**Universidad de las Ciencias Informáticas
FACULTAD 5**



**Definición y Aplicación de las Técnicas Adecuadas para el
Desarrollo de Requisitos en el Simulador Quirúrgico.**

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

Autor(es): Lisset Sigler Guerra.

Sailyn Salas Hechavarria.

Tutora: Lic. Yoisy Pérez Olmos.

Ciudad de La Habana

Julio del 2008

Es de importancia para quien desee alcanzar una certeza en su investigación, el saber dudar a tiempo.

Aristóteles.

DECLARACIÓN DE AUTORÍA.

Declaro que soy el único autor de este trabajo y autorizo al Simulador Quirúrgico 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 _____.

Autor: Sailyn Salas Hechavarria.

Autor: Lisset Sigler Guerra.

Tutor: Lic. Yoisy Pérez Olmos.

DATOS DE CONTACTO.

Lic. Yoisy Pérez Olmos: Licenciada en Ciencias de la Computación.
Correo: yoisy@uci.cu

*A mis **padres** por apoyarme y darme fuerza en mi vida.*

Por ser mi todo....

*A mi hermana **Rosy** y mi Abu **Bertha**, por ser las luces de
mi Universo.*

Sailyn.

*A mis **padres**, porque son lo más importante en mi vida.*

*A mis **abuelos**, para recompensar todo el cariño que me han dado.*

*A mi hermanito **Alejandro**, para que siga mis pasos,*

Para que vea en mí la inspiración.

*A **Yanko**, para que veas lo mucho que te quiero.*

Lisset.

AGRADECIMIENTOS

A quien será por siempre nuestro comandante “Fidel”, por construir esta revolución de la cual sentimos el inmenso orgullo de ser partícipe, por la oportunidad de estudiar en la UCI.

A nuestra tutora Yoisy por saber guiarnos y apoyarnos incondicionalmente en el transcurso de la elaboración de este trabajo, sin su ayuda no hubiese sido posible.

A Guille por estar siempre dispuesto a brindarnos su ayuda, por no saber decir “NO”.

A nuestros compañeros de grupo, porque de una forma u otra han influido en estos años de nuestra vida.

A los muchachos de proyecto, en especial a Leonel, Leandro y David.

A todos los profes, por transmitirnos sus conocimientos, por brindarnos su apoyo, por saber dar un consejo cuando fue necesario.

*A todos, Gracias
Sailyn y Lisset.*

A las dos personas más hermosa del mundo “Mis Padres”, que han sabido ser el centro de mi corazón. Gracias por confiar en mí en todo momento; a ellos les debo todo lo que soy.

A mi hermana Rosy por ser para mí un ejemplo a seguir. Te quiero mucho.

A mi abuela Bertha por mimarme tanto y sobre todo por tenerme presente en sus oraciones.

A mi compañera de tesis Lisset por soportarme estos tres años.

A mi familia por estar en los momentos más duro y alegres de mi vida.

A Dania Souchay, porque más que una profe es una gran “AMIGA”.

A Niobis por estar siempre ahí y brindarme su mano.

A mis compañeras de cuarto de 5to año Eli y Oda por haberme permitido ser parte de esa gran amistad.

A Yovanotti, por su paciencia y sus consejos.

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

*A todos, Gracias.
Sailyn.*

AGRADECIMIENTOS

A mis padres, por la confianza que han depositado en mí, por el apoyo que han sabido brindarme, por desempeñar con amor y ternura la difícil tarea de educarme y formarme como una persona de bien, ustedes han sido los mejores profes, la mejor escuela, los mejores amigos.

A mis abuelos Carmen y Rogelio por la preocupación de todos estos años, por tener siempre presente a la “niña”, aunque hace tiempo que deje de ser tan niña

A mi abuela Nana, por el amor y el cariño, por ser otra madre para mí.

A mi hermanito Ale, por existir.

A primo Robin, por ser más que un hermano.

A mis tíos Carmen Idea y Rogelio, por quererme tanto.

A los bichitos de la casa Christopher y Ogandy, aunque no me dejen dormir los fines de semana, los quiero mucho y sé que ustedes también me quieren a mí.

A Yanko, por el amor y el cariño, por permitirme formar parte de su vida, por ser un brazo firme en el cual apoyarme en los momentos en los que me sentía deprimida, por la experiencia vivida.

A Yiyi por apoyarme todo este tiempo, por preocuparse por mí.

A toda la familia en general, porque sé que todos se preocupan por mí.

A Dulce y Sarita, por un tercer año tan fenomenal, por su amistad incondicional.

A las mimis Aleamna y Liset, las quiero mucho.

A Rocny y Lanny por ser los mejores amigos.

A Ale, por regalarme una amistad tan especial.

A la profe Zoraida, por ser tan buena educadora, por haber sido como una madre para todos, porque sé que nos quiere como a sus propios hijos.

A mis compañeros de grupo de primer año.

A todos los amigos, compañeros y personas con los que compartí una sonrisa, una tristeza, un pedacito de mi corazón.

A Jana y Shaolín por alegrarme la vida.

A todos, Gracias.

Lisset.

RESUMEN.

El presente trabajo cuenta con un estudio del estado del arte enmarcado en elementos relacionados con el desarrollo de simuladores quirúrgicos y la Ingeniería de Requisitos, resumiéndose los principales problemas detectados en esta área en la construcción de simuladores cubanos. Se realizó además un estudio minucioso de las posibles técnicas a utilizar en cada una de las etapas definidas en el desarrollo de requisitos (elicitación, análisis, especificación y validación), con el objetivo de definir una propuesta adecuada para el Simulador Quirúrgico que tribute a la satisfacción del cliente, teniendo en cuenta las particularidades que diferencian al simulador del resto de los sistemas software.

Asimismo la propuesta de requisitos formalizada en las primeras etapas del desarrollo de requisitos se valida haciendo uso de la técnica de prototipado planteada en la etapa de validación obteniéndose un prototipo no funcional del simulador donde se representan las funcionalidades a implementar. Se realizó además una valoración de la propuesta teniendo en cuenta los criterios de los desarrolladores del proyecto y de algunos involucrados (Stakeholder).

PALABRAS CLAVES:

Simulador Quirúrgico (SimQ), Ingeniería de requisitos (IR), Técnicas, Desarrollo de requisitos, Prototipo no funcional.

TABLA DE CONTENIDO.

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	5
1.1. Estado del arte de Simuladores Quirúrgicos.	5
1.1.1. Desarrollo de Simuladores Virtuales en Cuba.	9
1.2. Requisitos de Software.	10
1.2.1. Tipos de Requisitos.	12
1.2.2. Principales características de los requisitos.	12
1.3. Importancia de los requisitos en la Industria del Software.	12
1.4. La Ingeniería de Requisitos.	17
1.5. Procedimientos usados en la IR.	18
1.6. Etapas del desarrollo de requisitos.	19
1.6.1. Elicitación de requisitos.	19
1.6.2. Análisis de requisitos.	22
1.6.3. Especificación de requisitos.	23
1.6.4. Validación de requisitos.	26
1.7. La IR en los Simuladores.	28
CAPÍTULO 2: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS	31
2.1. Estructura del proyecto Simulador Quirúrgico.	31
2.2. Propuesta de Elicitación de Requisitos para el SimQ.	32
2.3. Propuesta de Análisis de Requisitos para el SimQ.	34
2.3.1. Requisitos Funcionales del SimQ.	35
2.3.2. Requisitos No Funcionales del SimQ.	42
2.4. Propuesta de Especificación de Requisitos para el SimQ.	43
2.4.1. Especificación de los requisitos correspondientes a la primera y segunda etapa del SimQ. ...	44
CAPÍTULO 3: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA	71
3.1. Propuesta de Validación de Requisitos para el SimQ.	71
3.1.1. Prototipo No Funcional.	71
3.2. Valoración de la Propuesta de Solución Desarrollada.	81
3.2.1. Valoración por parte de los Desarrolladores.	81
3.2.2. Valoración por parte de los Stakeholders.	84
CONCLUSIONES	87

TABLA DE CONTENIDO

RECOMENDACIONES.....	88
REFERENCIAS BIBLIOGRÁFICAS.	89
BIBLIOGRAFÍAS CONSULTADAS.	92
ANEXOS.....	94
GLOSARIO DE TÉRMINOS.....	101
ÍNDICE DE FIGURAS.....	102
ÍNDICE DE TABLAS.	104

INTRODUCCIÓN

La idea de “mirar dentro” del cuerpo humano a través de un tubo - concepto con el cual nació la endoscopia - data de los siglos I y II al proponer un médico cirujano árabe, *Albukasin (936-1013)* la revisión de una cavidad interna. Durante varios siglos han sido muchas las dificultades tecnológicas, relacionadas con la calidad de la obtención de la imagen y la transmisión de la luz, que han debido superarse y que permitieron el desarrollo de procedimientos terapéuticos endoscópicos fundamentalmente en las especialidades de gastroenterología, urología y ginecología.

Sin embargo, no es hasta finales de la década de los 80 del pasado siglo que comienza a desarrollarse en su máximo esplendor la Cirugía de Mínimo Acceso (CMA), también conocida como cirugía video-endoscópica, debido a la incorporación de la tecnología del vídeo a la endoscopia.

Esta nueva forma de hacer cirugía, unida a consideraciones éticas relacionadas con las consecuencias que tienen para los pacientes los errores asociados a un adiestramiento insuficiente durante la curva de aprendizaje de los cirujanos, han implicado un cambio en los conceptos actuales de aprendizaje y enseñanza en estas técnicas.

El tradicional método de aprendizaje en cirugía, conocido como método de compañerismo (“mirar”-“hacer”-“enseñar”), en el que el cirujano en formación observa al profesor, luego realiza sus operaciones reproduciendo los gestos bajo supervisión, y posteriormente enseña a un nuevo estudiante, está siendo reemplazado por modelos de aprendizaje superiores (“mirar”-“practicar con seguridad”-“hacer”-“enseñar”), que insertan el concepto de “práctica segura” mediante la reproducción en un laboratorio de los gestos para adquirir las habilidades quirúrgicas, tantas veces como sea necesario, antes de operar un ser humano.

Estos modelos de aprendizaje actuales incluyen el empleo de maquetas, simuladores diversos, y el entrenamiento con animales, hasta llegar al empleo de la realidad virtual, que posibilita entrenar un mayor número de educandos en un menor tiempo y espacio, con costos de mantenimiento mínimos. Resaltando además la capacidad de los simuladores virtuales de evaluar objetivamente y medir el nivel de competencia adquirido por los alumnos.

En Cuba, después de más de 10 años de programa y aplicando las técnicas de aprendizaje tradicionales, el número de especialistas entrenados en procedimientos endoscópicos es muy bajo y no supera el 10%, e incluso en la mayoría de las especialidades la enseñanza de la CMA no está contemplada aún en sus programas de formación de residentes. Fácilmente se comprende la necesidad de buscar soluciones eficientes que estén a tenor con el desarrollo tecnológico actual y que permitan revolucionar metodológica y científicamente el proceso de enseñanza en la educación

quirúrgica mediante la aplicación de técnicas modernas de aprendizaje, tales como la simulación virtual.

Es por ello que con la experiencia en el tema de Realidad Virtual (RV) que viene desarrollando la Universidad de Ciencias Informáticas (UCI) se ha propuesto hacer realidad la idea de crear un Simulador Quirúrgico (SimQ), el cual se insertaría en el proceso de la enseñanza de la cirugía endoscópica formando parte además, del reducido número de simuladores que de este tipo han sido validados a nivel mundial.

La ingeniería de requisitos. Enfoque para el SimQ.

En la actualidad el proceso de desarrollo de muchos software presentan carencias metodológicas propias de una industria tan reciente como es la del software. Una de las principales dificultades, fundamentalmente en el desarrollo de sistemas de gran complejidad algorítmica, radica en que el equipo de desarrollo no tiene una idea clara de lo que se necesita construir. Existen una serie de dificultades que todos los proyectos de desarrollo de software presentan en el proceso de definición en cuanto a sistema se refiere, tal es el caso en que los propios usuarios no tienen una idea exacta de las necesidades globales de la organización. La aproximación tradicional a este problema en la industria de software ha sido asignar intermediarios (analistas) para obtener una lista de requisitos de cada usuario y de esta forma lograr una visión de conjunto y componer una especificación de requisitos completa, correcta y consistente (Jacobson, y otros, 2000).

La industria, en aras del mejoramiento de estos procesos, ha ido consolidando una nueva rama de la Ingeniería de Software (IS) que se ocupa de la investigación-desarrollo en el campo de los requisitos de software, rama denominada Ingeniería de Requisitos (IR). Como todas las demás actividades de la Ingeniería del software, debe adaptarse a las necesidades del proceso, el proyecto, el producto y las personas que realizan el trabajo (Pressman, 2006).

La IR proporciona el mecanismo apropiado para entender lo que el cliente quiere, analizar las necesidades, evaluar la factibilidad, negociar una solución razonable, especificar la solución sin ambigüedades, validar la especificación, y administrar los requisitos conforme estos se convierten en un sistema operacional (Pressman, 2006).

El proyecto Simulador Quirúrgico se encuentra en la primera etapa de su ciclo de vida, por lo que se hace visible la necesidad de un eficiente desarrollo de requisitos que posibilite definir lo que realmente se pretende realizar. Los simuladores virtuales en general, tienen particularidades que los diferencian del resto de las aplicaciones informáticas y que oscurecen el desarrollo de este proceso que tiene como objetivo fundamental definir claramente las funcionalidades que conformarán el sistema

propuesto. A continuación se especifican algunas de estas características especificadas para los simuladores quirúrgicos.

- Estrecha relación de trabajo entre dos disímiles campos de las ciencias como son la informática y la medicina: los ingenieros de software deben obtener y modelar la información y el conocimiento especializado de profesionales de la medicina.
- Gran complejidad y eficiencia algorítmica: aplicación de modelos matemáticos y/o físicos diseñados para lograr un mayor realismo en las escenas y la obtención de respuestas los más rápidas y eficientes posible.
- Modelos tridimensionales dinámicos con características específicas como el sonido: necesidad de la representación real del entorno a simular.
- Sensaciones de tacto en los componentes de hardware: respuesta emitida por el sistema.
- Posibilidad de uso de sistemas operativos Windows XP y GNU/Linux como plataforma de trabajo: Librería OpenGL (Open Graphics Library) multiplataforma como soporte gráfico a utilizar.

A partir de la problemática anterior surge el **problema de investigación** que da origen al presente trabajo de diploma: ¿Cómo definir y aplicar las técnicas adecuadas para el desarrollo de los requisitos en el proyecto Simulador Quirúrgico que guíe su proceso de desarrollo favoreciendo la entrega al cliente en tiempo y presupuesto?

El **objeto de estudio** de la investigación se basa en el Proceso de Ingeniería de Requisitos y se delimita como **campo de acción** el Proceso de Desarrollo de Requisitos en el Simulador Quirúrgico.

Para darle solución al problema planteado se propone como **objetivo general** definir y aplicar las técnicas adecuadas para el desarrollo de requisitos en el proyecto Simulador Quirúrgico que guíe su proceso de desarrollo favoreciendo la entrega al cliente en tiempo y presupuesto.

Para cumplimentar este objetivo general se definen los siguientes **objetivos específicos**:

1. Realizar el estado del arte acerca del desarrollo de los Simuladores Quirúrgicos y de la Ingeniería de Requisitos.
2. Investigar el proceso de desarrollo de requisitos en simuladores cubanos.

3. Definir las técnicas adecuadas a utilizar en cada una de las etapas que comprenden el desarrollo de requisitos para el Simulador Quirúrgico.
4. Aplicar la propuesta definida anteriormente en las dos primeras etapas de desarrollo del proyecto Simulador Quirúrgico.
5. Realizar una valoración de la propuesta teniendo en cuenta el criterio de desarrolladores del proyecto y de stakeholders.

Como **Idea a defender** se plantea que definiendo y aplicando las técnicas adecuadas para el desarrollo de los requisitos del Simulador Quirúrgico se obtendrá una especificación de requisitos que guíe su proceso de desarrollo favoreciendo la entrega al cliente en tiempo y presupuesto.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.

En este capítulo se abordará el estado del arte acerca de los Simuladores Quirúrgicos y de la Ingeniería de requisitos en general, señalándose la importancia de los mismos en la industria del software. También se especificarán las posibles técnicas a utilizar en cada una de las etapas comprendidas en el desarrollo de requisitos (elicitación, análisis, especificación y validación). Al finalizar el capítulo se resumirán los principales problemas detectados en el desarrollo de requisitos de simuladores cubanos a partir de datos reales obtenidos en una encuesta realizada a proyectos cubanos de simulación.

1.1. Estado del arte de Simuladores Quirúrgicos.

En la actualidad la expansión de las nuevas tecnologías de la información y las comunicaciones ha hecho necesario que se desarrolle a grandes escalas la rama de la informática. La informática gráfica ha venido destacándose, producto a la sensación virtual de la realidad que proporciona al hombre, durante las dos últimas décadas como la rama más sensacional de la informática. Un concepto a destacar en la informática gráfica es la Realidad Virtual (RV), donde se logran efectos de interactividad y realismo, alcanzando cambios en la forma de percibir el tiempo, el espacio e incluso la propia identidad. El término Realidad Virtual se relaciona al tratamiento de ambientes tridimensionales e interactivos orientados a la visualización de objetos 3D generados por computadoras, mostrando situaciones del mundo real en mundos virtuales (Jhon, y otros, 2006). Engloba herramientas que permiten visualizar un objeto tridimensional o un grupo de objetos tridimensionales, que en conjunto forman el entorno virtual, de modo que el usuario puede manipular, mediante algún tipo de dispositivo, una o varias características de los objetos y esto se visualice de forma interactiva y en tiempo real, creando la ilusión de que se forma parte del entorno. Los avances tecnológicos de los últimos años han posibilitado el desarrollo de la RV de forma acelerada. La Simulación se ha convertido en una técnica eficaz que permite reproducir los objetos reales cuando por problemas de tiempo, recursos o seguridad no es posible realizar la actividad en su medio natural, posibilitando el trabajo en campos o a distancias que antes impedían cualquier tipo de avance o comunicación (García, y otros, 2005). Grandes son los beneficios que brindan los Simuladores Virtuales (SV) en las diferentes ramas como por ejemplo, en la industria del entretenimiento (videojuegos), automotriz (simulador de carro), en la medicina (realización de prótesis, cirugías mínimamente invasivas, ingeniería genética), en la meteorología (estudio de tormentas eléctricas), paseos virtuales o en lo referido a la industria militar

desarrollándose habilidades mediante una ambientación digital con un gran ahorro de recursos y entrenamiento (Pacheco, y otros, 2007).

El desarrollo de la RV ha posibilitado un avance significativo específicamente en el ámbito de la medicina. Disímiles son los Simuladores Virtuales (SV) que se han creado en esta área, los cuales facilitan la manipulación de órganos internos del cuerpo en intervenciones quirúrgicas con una notable disminución de riesgo para el paciente, permiten además el entrenamiento de los médicos cirujanos en las habilidades necesarias para enfrentarse a una cirugía real.

Simuladores Quirúrgicos.

Los simuladores quirúrgicos son modelos virtuales de distintas partes del cuerpo dotados de herramientas de hardware con las que los cirujanos pueden entrenarse en distintas técnicas quirúrgicas mediante la utilización de entornos virtuales (*Ver Figura 1.1*) (Endosurgery, 2004).



Figura 1.1: Hardware de un Simulador Quirúrgico.

Los principales esfuerzos en el desarrollo de simuladores quirúrgicos están centrados en la cirugía mínimamente invasiva por la necesidad de simplificar su aprendizaje, debido a que este tipo de técnicas permiten reducir considerablemente el daño causado a los pacientes durante la intervención, acortando el periodo de convalecencia, disminuyendo el dolor postoperatorio, y los ratios de infección (Monserrat, y otros, 2004).

Este tipo de simuladores requieren que (Monserrat, y otros, 2004):

- Los objetos, órganos internos del paciente, se visualicen de la forma más realista posible.
- Los órganos respondan, de forma realista y en tiempo real, a las interacciones de los usuarios a partir de su deformación y a las restricciones existentes en la realidad.
- Los órganos también respondan, mediante modificaciones estructurales realistas, a acciones típicas quirúrgicas como cauterización, corte o sutura.

En función de los avances aplicados en el desarrollo de Simuladores Quirúrgicos y de los objetivos que éstos persiguen, se pueden agrupar en tres generaciones tecnológicamente secuenciales (Montserrat, y otros, 2004):

- Generación formada por los Simuladores Quirúrgicos que únicamente consideran la naturaleza geométrica de la anatomía humana.
- Generación que está constituida por aquellos que además, permiten la interacción física con las estructuras anatómicas.
- Generación que además de las características ya mencionadas de la segunda generación, tienen en cuenta la naturaleza funcional de los órganos.

A continuación se presentan algunos ejemplos de SimQ desarrollados mundialmente.

- **ProMIS:** Es uno de los simuladores existentes en el mercado. Permite a los cirujanos entrenarse en modelos virtuales y físicos en una misma unidad. Utiliza instrumentos y materiales quirúrgicos reales. El simulador ofrece un modelo físico en el que pueden entrenarse en la colocación de trocares y el uso de instrumental. Los instrumentos pueden ser intercambiados, retirados y reinsertados. Las señales de audio contribuyen a que la experiencia sea más realista. Los cirujanos también pueden entrenar en equipo. Cada módulo de aprendizaje de ProMIS se divide en distintas tareas y niveles, lo que proporciona una trayectoria clara en el aprendizaje del usuario. ProMIS analiza, entre otras cosas, el tiempo empleado, las longitudes de las trayectorias o la suavidad de los movimientos del cirujano: en qué grado sus movimientos son fluidos o erráticos. El simulador se integra con los planes de estudios existentes, pero puede ser modificado según necesidades particulares (Endosurgery, 2004).
- **GI Mentor:** Es otro modelo de simulador que incluye (*Ver Figura 1.2*) (Endosurgery, 2004):
 - Maniqués especialmente diseñados.
 - Programa de simulación.
 - Endoscopio real.
 - Accesorios endoscopios reales.
 - Programa de entrenamiento adaptable.

- Casos con escenarios reales de pacientes.

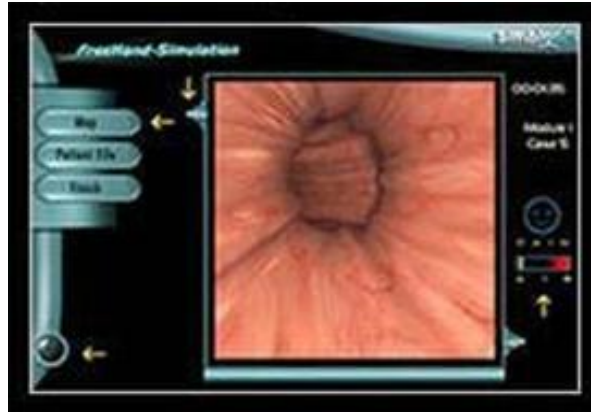


Figura 1.2: Pantalla del Simulador GI Mentor con la que entrenan los cirujanos.

- **LapSim System:** El entrenamiento de las habilidades básicas incluye la navegación con cámara, la navegación con instrumental, la coordinación, grapado, disección, colocación de clips, sutura y medición de precisión y velocidad. En todos los ejercicios el cirujano debe identificar el objeto propuesto y dirigir hacia él el instrumental, realizando la tarea requerida con la mayor precisión y en el menor tiempo posible (*Ver Figura 1.3*) (Endosurgery, 2004).

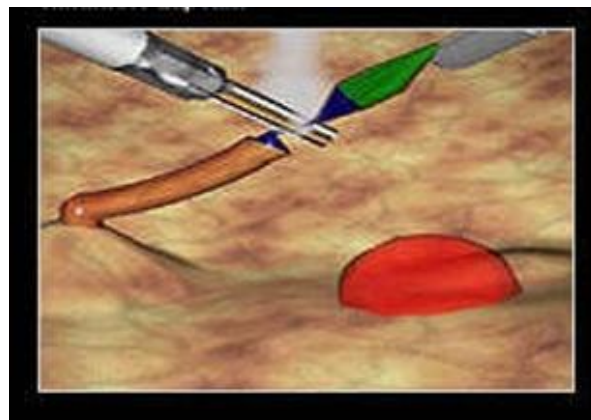


Figura 1.3: Imagen de Intervención a través del simulador LapSim.

- **Simbionix:** es una compañía productora de Simuladores médicos que recientemente desarrolló un nuevo simulador de endoscopia basado en la producción de un modelo tridimensional geométrico y posee una textura grabada en video durante un verdadero procedimiento endoscópica a través del cual los aprendices pueden realizar un cierto número de procedimientos endoscópicos hasta alcanzar la capacidad necesaria para enfrentarse a una

cirugía real. Este entrenamiento reduce el número de errores críticos que amenazan vida potencialmente (Stern, y otros, 2007).

- **Tower Trainer:** Utiliza tecnología de imagen patentada que permite el retrato realista. Es usado principalmente en el entrenamiento de la disección, sutura y nudo. El periscopio ajustable le permite al usuario sentarse o estar de pie y puede modificarse para encajar cualquier altura. El espejo superior simula a un amonestador video, llevando una repetición realista de una cirugía, y manipulación de instrumentos. Puede usarse con cualquiera de los modelos de la simulación (Ver Figura 1.4) (Corporation, S, 2004).



Figura 1.4: Simulador Tower Trainer.

1.1.1. Desarrollo de Simuladores Virtuales en Cuba.

El desarrollo científico alcanzado por Cuba y la necesidad de avanzar aceleradamente en el uso de las nuevas tecnologías de la información y las comunicaciones, ha provocado una evolución cuantitativa y cualitativa en la rama de la Informática Gráfica. Ejemplo de esto es la concepción y el desarrollo de espacios virtuales en Internet para la ciencia, encaminados a promover el trabajo colaborativo entre comunidades científicas y académicas, tomando como base la infraestructura técnica y organizativa existente en el país y su proyección para los próximos años (Ibarreche, y otros, 2007).

A mediados de la década del noventa se hizo necesario buscar alternativas dirigidas a elevar la preparación combativa de las tropas con un mínimo de recursos, y sustituir con medios propios los antiguos equipos entrenadores electromecánicos soviéticos, deteriorados por las carencias provocadas

por el periodo especial. Es por ello que los especialistas del Centro de Investigación y Desarrollo de Simuladores (SIMPRO), perteneciente a las FAR, en coordinación con la industria militar cubana y con otras entidades de la economía nacional se trazan como reto la producción de simuladores virtuales que solucionaran la problemática existente. Una muestra de ello lo constituyen varias decenas de simuladores virtuales existentes en el país, diseminados por unidades de las FAR, escuelas de automovilismo del Ministerio de Transporte, y de patrullas del Ministerio del Interior. Unos permiten el entrenamiento para conducción de tanques, carros blindados y de distintos vehículos; otros, el tiro cohetil, de infantería, artillería y de tanques; algunos, los más complejos, capacitan a pilotos de aeronaves de transporte y de combate. Todos son diseñados y fabricados en Cuba.

Los soldados de las Fuerzas Armadas revolucionarias (FAR) hoy en día pueden realizar sus prácticas en el simulador de tiro de infantería posibilitando el ahorro de municiones y la creación de habilidades sin correr peligro alguno, lo que constituye un modelo a escala real que reproduce íntegramente la cabina de los tanques como vehículos de combate, no solo en estructura y accesorios (pedales, palancas y otros.). En la pantalla del conductor se presentan diferentes relieves del terreno a transitar, condiciones climatológicas, movimientos y sonidos característicos de este medio de combate, y obstáculos a vencer: desde puentes, túneles y asentamientos, hasta campos minados y zanjas antitanques (Pérez, 2007).

En el año 2003 la Empresa SIMPRO en unión con la Universidad de Ciencias Informáticas hicieron posible la realización de un simulador de conducción. El aspirante, mediante su entrenamiento en una cabina real que transita por una ciudad virtual, puede obtener las habilidades necesarias para conducir. Esta nueva oportunidad elimina los riesgos de accidentes automovilísticos durante una fase de aprendizaje, además del ahorro de combustible necesario para el entrenamiento.

En el área de la medicina aún no se tienen resultados en Cuba, por lo que el desarrollo del simulador que da lugar al presente trabajo de diploma constituye la primera experiencia de este tipo.

1.2. Requisitos de Software.

Uno de los párrafos más citados en la bibliografía de la Ingeniería del Software dice: "La parte más difícil de construir un sistema es precisamente saber qué construir. Ninguna otra parte del trabajo conceptual es tan difícil como establecer los requerimientos técnicos detallados, incluyendo todas las interfaces con personas, máquinas y otros sistemas. Ninguna otra parte del trabajo afecta tanto el sistema si no se realiza correctamente. Ninguna es tan difícil de corregir más adelante. Entonces, la

tarea más importante que el Ingeniero de Software hace para el cliente es la extracción iterativa y el refinamiento de los requerimientos del producto" (Brooks, 1987). Precisamente uno de los problemas que persiste en la industria del software actualmente está dado por un inadecuado entendimiento de las necesidades de los usuarios, la incapacidad de absorber cambios en los requerimientos y las insatisfacciones de los clientes por inaceptable o bajo desempeño del software. Algunos investigadores de la industria de software han utilizado diferentes definiciones para representar verbalmente el concepto de requisito.

- La IEEE Standard Glossary of Software Engineering Terminology, define los requerimientos como (Chaves, 2006):
 1. Condición o capacidad que necesita un usuario para resolver un problema o lograr un objetivo.
 2. Condición o capacidad que tiene que ser alcanzada o poseída por un sistema o componente de un sistema para satisfacer un contrato, estándar, u otro documento impuesto formalmente.
 3. Una representación documentada de una condición o capacidad como en 1 o 2.

En esta definición no se tratan los requisitos en función de la satisfacción de clientes y usuarios.

- Craig Larman define los requisitos como una descripción de las necesidades o deseos de un producto y delimita la meta primaria de la fase de requerimientos en la identificación y documentación de lo que realmente se necesita, en una forma que claramente se lo comunique al cliente y a los miembros del equipo de desarrollo (Larman, 1999).

En esta definición sí se tiene en cuenta a los requisitos como forma de entendimiento entre los desarrolladores y los clientes.

- Según Ian Sommerville un requerimiento de software es simplemente una declaración abstracta de alto nivel de un servicio que debe proporcionar el sistema o una restricción de éste (Chaves, 2006).

Después de analizar las diferentes definiciones de requerimientos citadas anteriormente en la presente investigación se define un requisito como las *"características, cualidades, condiciones y capacidades de un sistema descritas claramente tanto para el equipo de desarrollo como para clientes y/o usuarios en función de satisfacer sus necesidades"*.

1.2.1. Tipos de Requisitos.

Los requisitos se pueden clasificar en dos categorías: requisitos funcionales y requisitos no funcionales. Los funcionales especifican acciones o funciones que el sistema debe ser capaz de realizar, describen las transformaciones que el sistema realiza sobre las entradas para producir salidas. Es importante que se describa el ¿Qué? y no el ¿Cómo? se deben hacer esas transformaciones. Estos requerimientos al tiempo que avanza el proyecto de software se convierten en los algoritmos, la lógica y gran parte del código del sistema. En el caso de los no funcionales detallan las propiedades o características que hagan al sistema más atractivo y usable y generalmente se distribuyen en diferentes categorías como el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares entre otros., tales como la fiabilidad y la seguridad (Chaves, 2006).

1.2.2. Principales características de los requisitos (Chaves, 2006).

- **Conciso:** Un requerimiento es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.
- **No ambiguos:** Un requerimiento no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.
- **Consistente:** Un requerimiento es consistente si no es contradictorio con otro requerimiento.
- **Completos:** Un requerimiento está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.
- **Realistas:** Puede el sistema hacer lo que el cliente desea.
- **Posible de probar o verificar.** Si un requerimiento no se puede comprobar, entonces ¿cómo se sabe si se cumplió con él o no? Un requisito es verificable si existe algún proceso finito de coste razonable que pueda probar que el producto software cumple con el requisito.

1.3. Importancia de los requisitos en la Industria del Software.

Según (Wiegers, 2003) los errores durante la etapa de requisitos representan del 40 al 60 % de todos los defectos encontrados en un proyecto de software. Los dos problemas reportados más frecuentemente en una revisión de la industria de software europea conciernen a la especificación y administración de los requerimientos del cliente. Sin embargo muchas organizaciones todavía

practican métodos ineficientes para estas actividades esenciales de proyecto. El resultado típico es la diferencia marcada entre lo que los desarrolladores suponen que desea el cliente y lo que realmente estos necesitan.

Otros informes del Instituto de Ingeniería de Software, (SEI) por sus siglas en inglés: *Software Engineering Institute*, plantean que alrededor de un 50% de los problemas que sufren los proyectos de software están originados en una deficiente etapa definición y gestión de requisitos. Sobre el 45% del esfuerzo total de un proyecto se debe a costos de retrabajo debido también a requisitos defectuosos. Un problema en la fase de requisitos, si no es encontrado a tiempo, puede costar entre 5 y 200 veces más esfuerzo solucionarlo que si se hubiese detectado a tiempo.

En (Duran, 2002) se tienen algunos datos sobre los resultados en la producción de proyectos de software:

Desde 1968 se ha invertido un gran esfuerzo en determinar las causas y proponer soluciones para la crisis del software. En 1979, la oficina de Cuentas del gobierno norteamericano (GAO), realizó un estudio a 9 proyectos de desarrollo de software para el gobierno norteamericano cuyos contratos sumaban una cantidad total 6.800.000 dólares. De esta cantidad, solo 119.000 dólares correspondían a un proyecto que se había utilizado tal y como se había integrado. Dicho proyecto se trataba sobre un procesador de COBOL, por lo que era un problema relativamente simple cuyos requisitos eran comprendidos por clientes y desarrolladores y que además no cambiaron durante el desarrollo (Ver *Figura 1.5*).

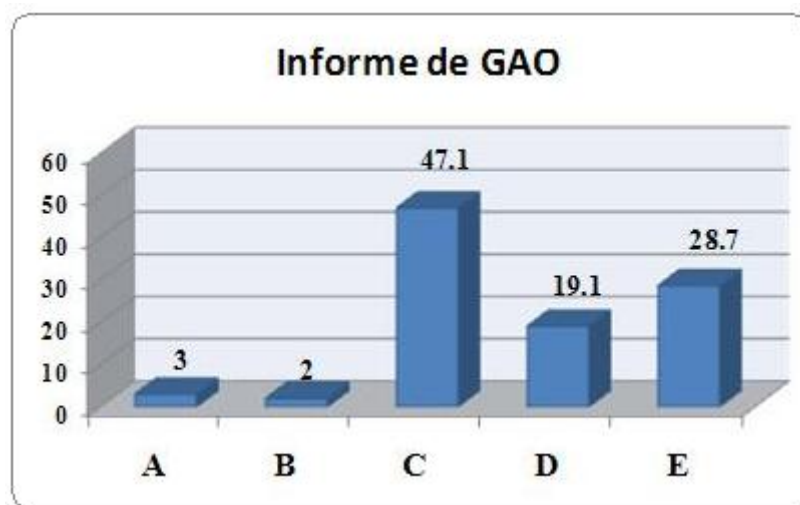


Figura 1.5: Resultados del informe de GAO.

- A:** Usado después de cambios (3%)
- B:** Usado tal como se entregó (2%)
- C:** Entregado pero nunca usado (47.1%)
- D:** Usado pero ampliamente reelaborado o abandonado después. (19.1%)
- E:** Pagado pero no entregado. (28.7%)

En 1995, el *Grupo Standish* de Estados Unidos (EEUU) realiza un estudio (el informe CHAOS) mucho más amplio y significativo que el de GAO, cuyos resultados, a pesar de haber pasado más de 25 años, no reflejaban una mejoría sustancial (Ver Figura 1.6).

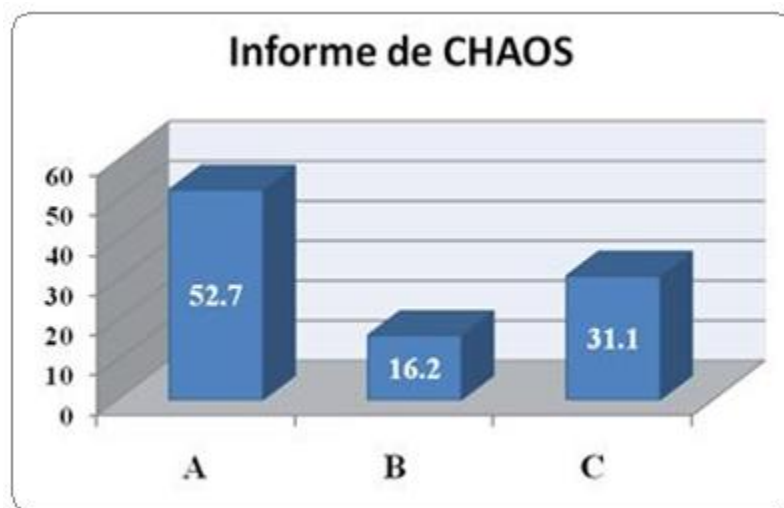


Figura 1.6: Resultados del informe de CHAOS.

- A:** Terminado y operativo pero fuera de plazo, presupuesto y sin satisfacer todos los requisitos. (52.7%)
- B:** Terminado dentro del plazo y presupuesto cumpliendo todos los requisitos. (16.2%)
- C:** Cancelado durante el desarrollo. (31.1%)

Sin incluir el 16.2% de los proyectos terminados correctamente, la media del gasto final fue del 189% del presupuesto original, el tiempo necesario para su realización del 222% del plazo original y se cumplieron una media del 61% de los requisitos iniciales, cifras que también empeoraban en caso de grandes compañías.

En las encuestas realizadas a los directivos de los proyectos que participaron en el estudio, éstos indicaron que en su opinión los principales factores de éxitos eran:

- Inclusión de los usuarios.
- Apoyo de los directivos.
- Enunciado claro de los requisitos.

Como factores de fracaso identificaron:

- Falta de información por parte de los usuarios.
- Especificaciones y requisitos incompletos.
- Especificaciones y requisitos cambiantes.

Los resultados de estos informes muestran que casi un tercio de los proyectos de desarrollo de software se cancelan durante su desarrollo y que la gran mayoría presenta graves desviaciones respecto a plazos y presupuestos iniciales.

Además, las principales causas, tanto de éxito como de fracaso, indicadas por los directivos encuestados en los informes CHAOS están relacionadas directamente con la correcta identificación y gestión de los requisitos que debían cumplir las aplicaciones desarrolladas para satisfacer las necesidades de los usuarios.

En (Rangel, 2007) se corrobora lo planteado anteriormente al señalar que en las investigaciones realizadas por el Grupo *Standish* en EEUU se obtuvo que el 31.1% de los proyectos de software son cancelados antes de que se completen. El 52.7% costarán 189% más de sus estimaciones iniciales. El 75 % de los productos de software grandes se entregaron a los clientes pero tienen fallas, son un fracaso porque no se usan o no cumplen los requerimientos del cliente. Se añade además el origen de los errores de los proyectos de software (Ver Tabla 1.1) (Ver Figura 1.7).

Tabla 1.1: Origen de los errores de los proyectos de software.

Fase	Por ciento
Obtención y análisis de requerimientos	56
Diseño	10
Codificación	7
Otros	27

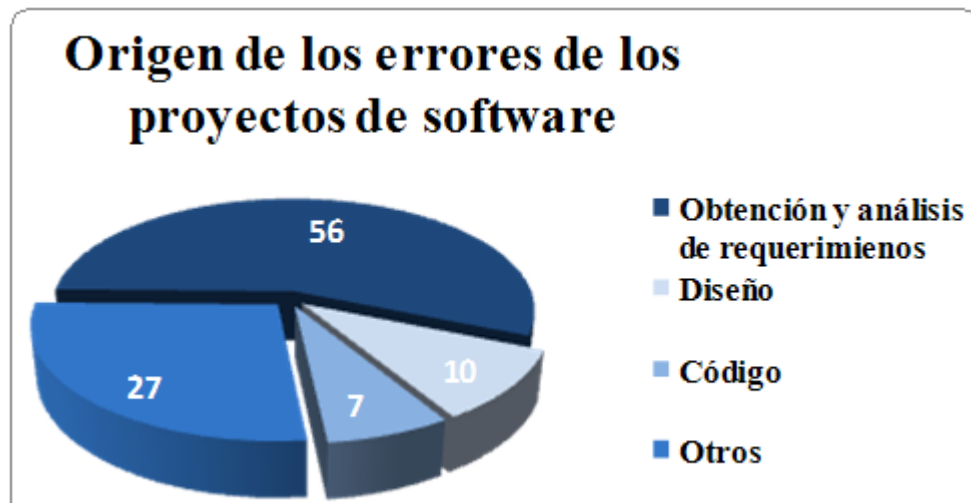


Figura 1.7: Origen de los errores de los proyectos de software.

En 1996, el proyecto ESPITI (*European Software Process Improvement training Initiative*) realiza una investigación sobre los problemas en el desarrollo del software a nivel de Europa. Los resultados indicaron que los mayores problemas estaban también relacionados con las especificaciones y la ingeniería y gestión de los requisitos.

Otro hecho comprobado es que el costo de un cambio en los requisitos, una vez entregado el producto, es entre 60 y 100 veces superior al costo que hubiera representado el mismo cambio durante las fases iniciales de desarrollo, por lo que no es lógico que los proyectos en los que no se determinan correctamente los requisitos y cambian frecuentemente durante el desarrollo, superan con creces su presupuesto iniciales (Duran, 2002).

Según Lizka Johany Herrera en su documento "Ingeniería de Requerimientos- Ingeniería de Software", la IR aporta variados beneficios en el desarrollo de software (Chaves, 2006):

- Permite gestionar las necesidades del proyecto en forma estructurada: Cada actividad de la IR consiste de una serie de pasos organizados y bien definidos.
- Mejora la capacidad de predecir cronogramas de proyectos, así como sus resultados: La IR proporciona un punto de partida para controles subsecuentes y actividades de mantenimiento, tales como estimación de costos, tiempo y recursos necesarios.
- Disminuye los costos y retrasos del proyecto: es sabido que reparar errores por un mal desarrollo no descubierto a tiempo, es sumamente caro; especialmente aquellas decisiones

tomadas durante la IR, ya que es una de las etapas de mayor importancia en el ciclo de desarrollo de software y de las primeras en llevarse a cabo.

- Mejora la calidad del software: La calidad en el software tiene que ver con cumplir un conjunto de requerimientos (funcionalidad, facilidad de uso, confiabilidad, desempeño y otros).
- Mejora la comunicación entre equipos: La especificación de requerimientos representa una forma de consenso entre clientes y desarrolladores. Si este consenso no ocurre, el proyecto no será éxitos.
- Evita rechazos de usuarios finales: La ingeniería de requerimientos obliga al cliente a considerar sus requerimientos cuidadosamente y revisarlos dentro del marco del problema, por lo que se le involucra durante todo el desarrollo del proyecto.

1.4. La Ingeniería de Requisitos.

Muchos son los ejemplos y estadísticas que demuestran el alto costo de errores en las actividades relacionadas con los requisitos en los sistemas de software. Es por ello que en los últimos años, a la par del desarrollo de la industria de software, ha aumentado también el interés por adoptar las mejores prácticas en los requisitos de software. En este sentido ha surgido el término Ingeniería de Requisitos con el objetivo de englobar los procesos de gestión de requisitos en el ciclo de vida del software.

- Una definición algo antigua, pero todavía vigente es la citada por Boehm en 1979: Ingeniería de Requerimientos es la disciplina para desarrollar una especificación completa, consistente y no ambigua, la cual servirá como base para acuerdos comunes entre todas las partes involucradas y en donde se describen las funciones que realizará el sistema (Bahamonde, y otros, 2003).
- En (Presman, 2002) Pressman retoma que la IR facilita el mecanismo apropiado para comprender lo que quiere el cliente, analizando necesidades, confirmando su viabilidad, negociando una solución razonable, especificando la solución sin ambigüedad, validando la especificación y gestionando los requisitos para que se transformen en un sistema operacional.

Cualquiera que sea la definición usada para la IR debe señalarse su importancia en el proceso de desarrollo de software de un proyecto.

La Ingeniería de Requerimientos (IR) cumple un papel primordial en el proceso de producción de software, ya que se enfoca un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes; de esta manera, se pretende minimizar los problemas relacionados con la mala gestión de los requerimientos en el desarrollo de un sistema (Chaves, 2006).

Cabe a la Ingeniería de Requisitos, proponer métodos, técnicas y herramientas que faciliten el trabajo de definición de lo que se quiere de un artefacto de software (Ridao, y otros, 2000).

1.5. Procedimientos usados en la IR.

En (Pressman, 2005) se define el proceso de IR en siete áreas de esfuerzo: inicio, obtención, elaboración, negociación, especificación, validación y gestión de requisitos.

En (Wieggers, 2003) Karl E. Wieggers divide el dominio de la IR en:

- Desarrollo de requisitos.
- Administración de requisitos.

El desarrollo de requisitos es dividido a su vez en cuatro subprocesos (Ver Figura 1.8).

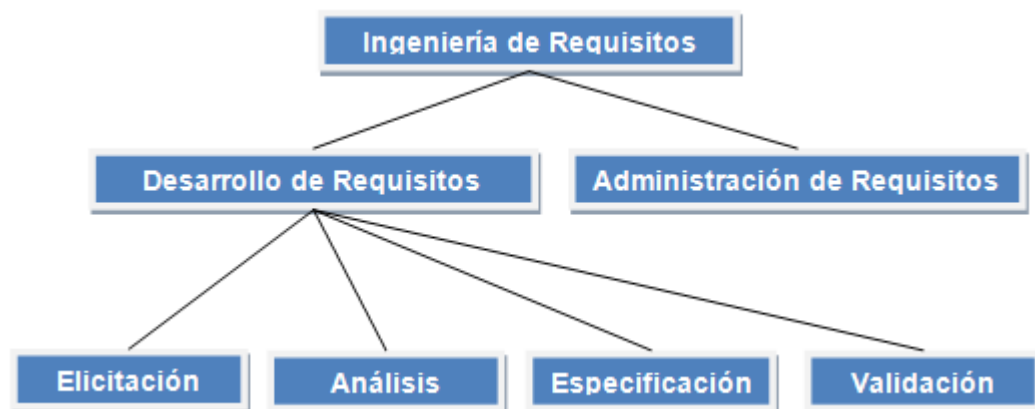


Figura 1.8: Subcomponentes de la IR.

En la (Chaves, 2006) Michael Arias también coincide y afirma que independientemente de que existen diferentes enfoques de procedimientos usados en la IR se tiene un denominador común que consiste en el desarrollo de requisitos y que puede resumirse en las siguientes etapas fundamentales: **Elicitación, Análisis, Especificación, Validación** (Ver Figura 1.9).

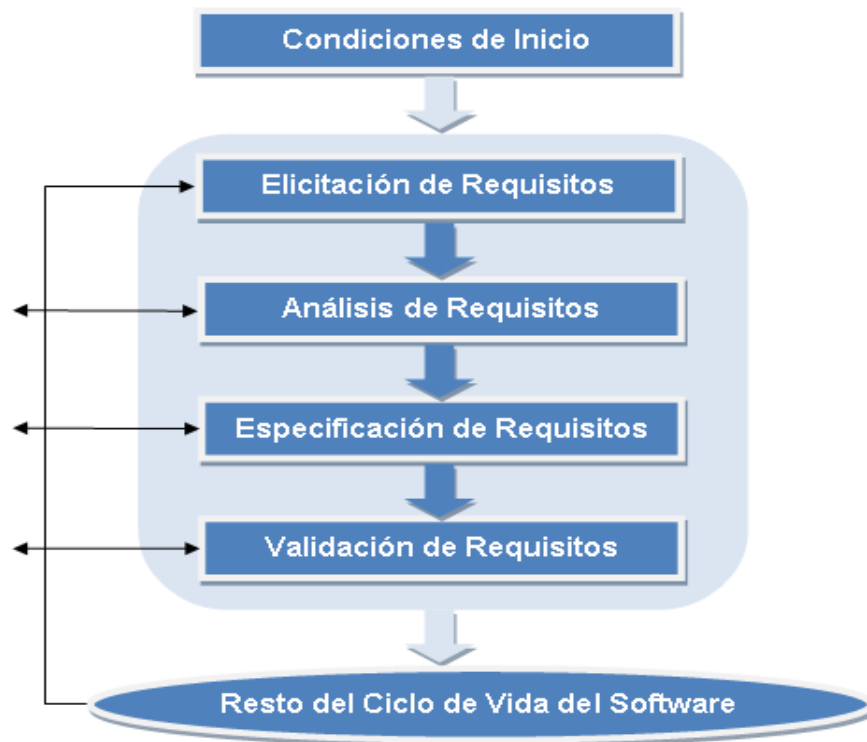


Figura 1.9: Procedimiento del desarrollo de requisitos.

1.6. Etapas del desarrollo de requisitos.

Retomando el procedimiento propuesto por Karl E. Wieggers citado en el epígrafe anterior que divide la IR en cuatro etapas que forman parte a su vez del desarrollo de requisitos como propuesta para el desarrollo de la presente investigación, en este tópico se detallará cada una de las etapas definidas.

1.6.1. Elicitación de requisitos.

La etapa de elicitación de requisitos abarca la primera y quizás más importante fase dentro del desarrollo de un sistema informático (Villanueva, y otros, 2005). En esta etapa se identifican los elementos de información que determinan las características deseadas y las restricciones que deberá satisfacer el sistema software que tendrán efectos satisfactorios para el usuario en el ambiente donde se encuentra. Se refiere a la captura y descubrimiento de los requisitos que deberán ser implementados (Borland, 2006).

En (Borland, 2006) se citan algunas de las actividades fundamentales de esta etapa:

- Definir la visión del producto y el alcance del proyecto.
- Identificar clases de usuarios y sus características.

- Identificar eventos y respuestas del sistema.

Teniéndose como objetivo principal de la fase la obtención de una especificación preliminar detallada (informal) de las necesidades de los usuarios del software a desarrollar (Ver Figura 1.10).

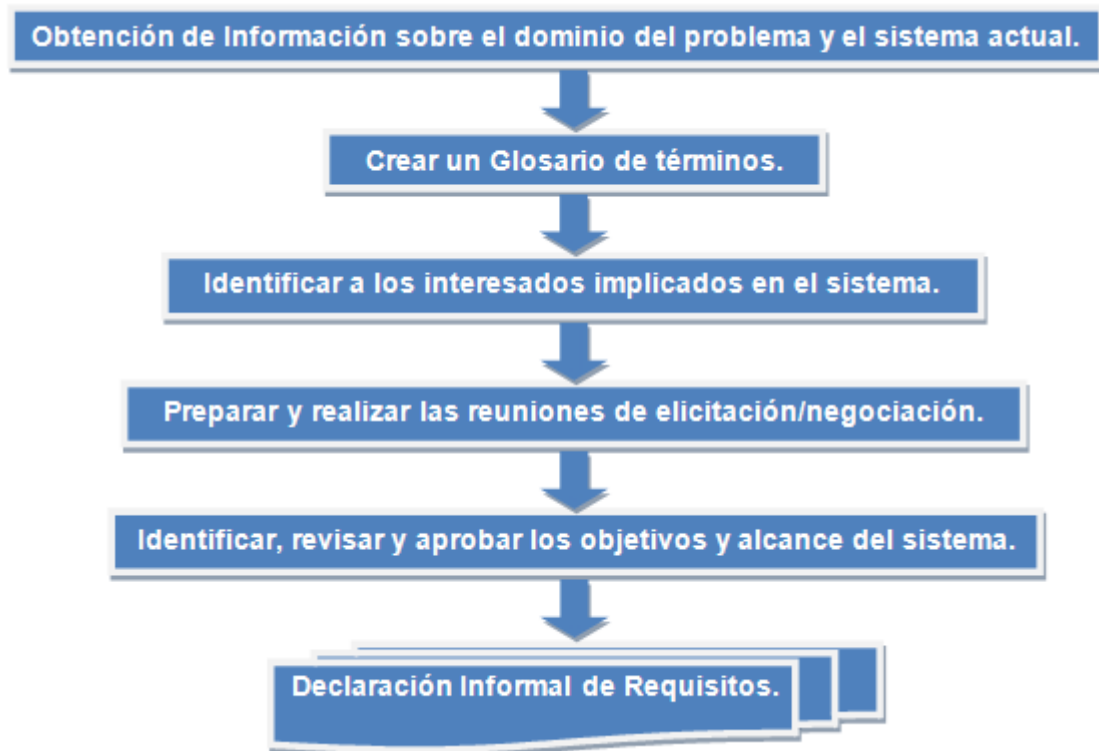


Figura 1.10: Procedimiento para la elicitación de requisitos.

Técnicas usadas en la elicitación para la obtención de requisitos.

- **Entrevistas:** Las entrevistas resultan una técnica muy aceptada dentro de la ingeniería de requisitos y su uso está ampliamente extendido. A través de esta técnica el equipo de trabajo se acerca al problema de una forma natural. Existen muchos tipos de entrevistas y son muchos los autores que han trabajado en definir su estructura y dar guías para su correcta realización. Básicamente, la estructura de la entrevista abarca tres pasos: identificación de los entrevistados, preparación de la entrevista, realización de la entrevista y análisis de los resultados.

La entrevista no es una técnica sencilla de aplicar, pues requiere que el entrevistador sea experimentado y tenga capacidad para elegir bien a los entrevistados para obtener de ellos toda la información posible en un período de tiempo siempre limitado (Escalona, y otros, 2002).

- **JAD (Joint Application Development / Desarrollo conjunto de aplicaciones):** Esta técnica es una práctica de grupo donde se desarrolla un conjunto de reuniones durante un período de 2 a 4 días en las que participan analistas, usuarios, administradores del sistema y clientes. En estos talleres se les ayuda a los clientes y usuarios a formular problemas y proponer posibles soluciones, involucrándolos y haciéndolos partícipes del desarrollo del proyecto (Escalona, y otros, 2002).
- **Brainstorming (Tormenta de ideas):** Tormenta de ideas es también una técnica de reuniones en grupo cuyo objetivo es que los participantes muestren sus ideas de forma libre. Consiste en la mera acumulación de ideas y/o información sin evaluar las mismas. Las sesiones de brainstorming deben estar integradas por un número de cuatro a diez personas, uno de los cuales debe asumir el rol de moderador de la sesión, pero sin carácter de controlador. Como técnica de captura de requisitos es sencilla de aplicar. Frente al JAD tiene la ventaja de que es muy fácil de aprender y requiere poca organización. Además suele ofrecer una visión general de las necesidades del sistema, pero normalmente no se obtienen resultados con el mismo nivel de detalle que en otras técnicas (Escalona, y otros, 2002).
- **Casos de Uso:** Aunque inicialmente se desarrollaron como técnica para la especificación de requisitos, algunos autores proponen casos de uso como técnica para la captura de requisitos. Los casos de uso permiten mostrar el contorno (actores) y el alcance (requisitos funcionales expresados como casos de uso) de un sistema. La ventaja esencial de los casos de uso es que resultan muy fáciles de entender para el usuario o cliente, sin embargo carecen de la precisión necesaria si no se acompañan con una información textual o detallada con otra técnica como pueden ser los diagramas de actividades (Escalona, y otros, 2002).
- **Cuestionarios y Lista de chequeo (Checklists):** Esta técnica requiere que el analista conozca el ámbito del problema en el que está trabajando. Consiste en redactar un documento con preguntas cuyas respuestas sean cortas y concretas, o incluso cerradas por unas cuantas opciones en el propio cuestionario (Checklist). Este cuestionario será cumplimentado por el grupo de personas entrevistadas o simplemente para recoger información en forma independiente de una entrevista (Escalona, y otros, 2002).
- **Mapas conceptuales:** Los mapas conceptuales son grafos en los que los vértices representan conceptos y las aristas representan posibles relaciones entre dichos conceptos. Estos grafos de relaciones se desarrollan con el usuario y sirven para aclarar los conceptos relacionados con el sistema a desarrollar. Son muy usados dentro de la ingeniería de requisitos, pues son fáciles de entender por el usuario, más aún si el equipo de desarrollo hace

el esfuerzo de elaborarlo en un lenguaje común. Sin embargo, deben ser usados con cautela porque en algunos casos pueden llegar a ser ambiguos en casos complejos, si no se acompaña de una descripción textual (Escalona, y otros, 2002).

- **Storyboards:** Esta técnica consiste en representar sobre papel, en forma muy esquemática, las diferentes interfaces al usuario. Estos esbozos pueden ser agrupados y unidos por enlaces dando idea de la estructura de navegación (guiones gráficos) (Escalona, y otros, 2002). Storyboards puede acelerar el desarrollo conceptual de diferentes facetas de una aplicación. Puede ser utilizado para entender reglas del negocio que serán implementadas, para definir algoritmos y otras construcciones matemáticas que deben ser ejecutadas en el sistema o para demostrar reportes u otras salidas impresas. Ciertamente storyboards pueden y deberían servir virtualmente para cualquier tipo de aplicación en la cual la retroalimentación temprana de los involucrados podría ser un importante factor de éxito (Leffingwell, y otros, 2003).
- **Aprendiz:** Esta técnica se basa en la idea del maestro y en la observación del trabajo real. El ingeniero de software representa al aprendiz y el usuario/cliente cumple el rol de maestro. El aprendiz se sienta con el maestro a aprender por medio de la observación, haciendo preguntas y también realizando algún trabajo bajo la supervisión del maestro.

Generalmente para el cliente/usuario es difícil explicar completamente su trabajo. En ocasiones tiende a omitir elementos que consideran triviales. La técnica del aprendiz es muy útil para mitigar este problema. Es también una técnica apropiada para un proyecto donde el problema no es estructurado, o sea, en casos en los que se necesita obtener conocimiento sobrentendido por los clientes/usuarios. Tiene como limitante que su implementación requiere de mucho tiempo (Robertson, y otros, 2006).

Durante el desarrollo de esta etapa de la IR se suele aplicar combinaciones de varias de estas técnicas.

1.6.2. Análisis de requisitos.

Es la actividad de la IR en la cual se estudia la información extraída durante la elicitación, para identificar la presencia de áreas no detectadas, requisitos contradictorios y peticiones que aparecen como vagas e irrelevantes (Ávila, y otros, 2007), así como otras deficiencias. Durante esta actividad se refinan los requerimientos para asegurar el buen entendimiento de estos por parte de todos los interesados, incluyendo la evaluación de la factibilidad de los requisitos (Borland, 2006).

El objetivo fundamental de esta etapa de análisis es descubrir problemas en la declaración informal de requisitos generados durante la captura de los mismos y se obtiene como resultado una lista formal de la definición de los requisitos del sistema a implementar (Ver *Figure 1.11*) (Ávila, y otros, 2007).

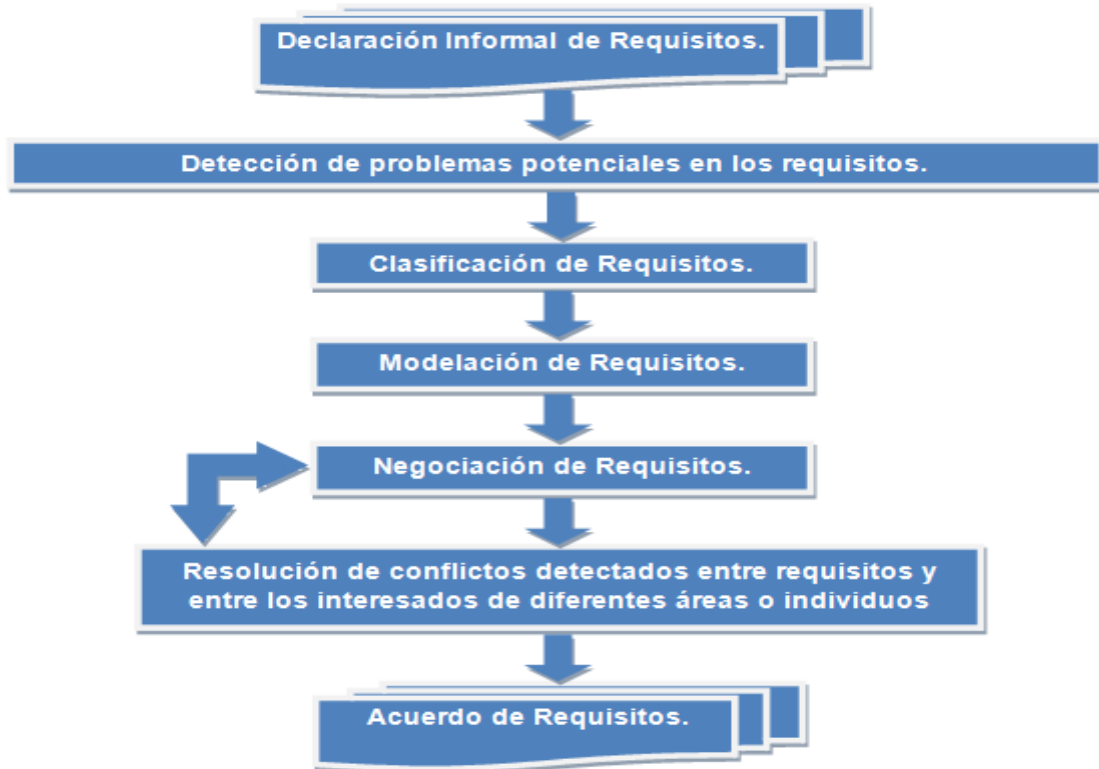


Figure 1.11: Procedimiento para el análisis de requisitos.

1.6.3. Especificación de requisitos.

La especificación de requisitos, conocida también como definición de requisitos, es el modo habitual de guardar y comunicar requisitos. El objetivo fundamental de esta actividad es obtener un documento de especificación: Especificación de Requisitos de Software (ERS) que defina, de forma completa, precisa y verificable, los requisitos que debe cumplir el sistema, tanto funcionales como no funcionales, así como las restricciones aplicables al diseño (software y hardware). Debe abordar la descripción de lo que hay que desarrollar, no el cómo ni el cuándo. No debe incluir requisitos innecesarios, no solicitados por el cliente, ni incluir detalles sobre el diseño del sistema (Ver *Figure 1.12*) (Ávila, y otros, 2007).



Figura 1.12: Procedimiento para la especificación de requisitos.

Técnicas de especificación de requisitos.

Las técnicas más comúnmente usadas para especificar los requisitos de las aplicaciones de software están basadas en lenguaje natural debido a que es fácilmente entendida por los involucrados en el proceso. No obstante existen casos en los que la ambigüedad de los lenguajes naturales no es tolerable, particularmente cuando los requisitos tratan asuntos tan delicados como la vida de un paciente o cuando el comportamiento erróneo de un sistema puede conducir a graves consecuencias financieras o legales (Pérez, 2007). En la literatura se reportan algunas técnicas que permiten especificar esos requisitos de una manera más técnica.

- **Lenguaje natural:** Resulta una técnica muy ambigua para la definición de los requisitos. Consiste en definir los requisitos en lenguaje natural sin usar reglas para ello. A pesar de que son muchos los trabajos que critican su uso, es cierto que a nivel práctico se sigue utilizando (Escalona, y otros, 2002).
- **Plantillas o patrones:** Esta técnica tiene por objetivo describir los requisitos mediante el lenguaje natural pero de una forma estructurada. Una plantilla es una tabla con una serie de campos y una estructura predefinida que el equipo de desarrollo va completando orientado al lenguaje del usuario. Las plantillas eliminan parte de la ambigüedad del lenguaje natural al

estructurar la información; cuanto más estructurada sea ésta, menos ambigüedad ofrece. Sin embargo, si el nivel de detalle elegido es demasiado estructurado, el trabajo de llenar las plantillas y mantenerlas, puede ser demasiado tedioso (Escalona, y otros, 2002).

- **Casos de uso:** Como técnica de especificación de requisitos es como más ampliamente han sido aceptados los casos de uso. Actualmente se ha propuesto como técnica básica del proceso RUP. Sin embargo, son varios los autores que hablan de la ambigüedad de los resultados en la definición de los requisitos, por lo que hay propuestas que los acompañan de descripciones basadas en plantillas (Escalona, y otros, 2002).
- **Lenguajes Formales:** Otro grupo de técnicas que merece la pena resaltar como extremo opuesto al lenguaje natural, es la utilización de lenguajes formales para describir los requisitos de un sistema. Las especificaciones algebraicas como ejemplo de técnicas de descripción formal, han sido aplicadas en el mundo de la ingeniería de requisitos desde hace años. Sin embargo, resultan muy complejas en su utilización y para ser entendidas por el cliente. El mayor inconveniente es que no favorecen la comunicación entre cliente y analista. Por el contrario, es la representación menos ambigua de los requisitos y la que más se presta a técnicas de verificación automatizadas (Escalona, y otros, 2002).
- **Historias de Usuarios (*User Stories*):** Las Historias de Usuarios son usadas para la especificación de requisitos. Cada Historia de Usuarios es una pequeña descripción del comportamiento del sistema, desde el punto de vista del usuario de sistema. Además es escrita en una pequeña tarjeta para asegurar que no sea demasiado grande (Fernández, 2002). Las escriben los propios clientes a partir de su visión acerca de las necesidades del sistema. El tratamiento de las historias de usuario es muy dinámico y flexible. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas (Canós, y otros, 2000).
- **Escenarios:** La técnica de los escenarios consiste en describir las características del sistema a desarrollar mediante una secuencia de pasos. La representación del escenario puede variar dependiendo del autor. Esta representación puede ser casi textual o ir encaminada hacia una representación gráfica en forma de diagramas de flujo. El análisis de los escenarios, hechos de una forma u otra, pueden ofrecer información importante sobre las necesidades funcionales de sistema (Escalona, y otros, 2002). Existen varios estilos para construir escenarios, tales como narrativa textual, storyboards, videos y prototipos escritos (Hadad, et al., 1999).

El modelo de escenario que se utiliza comúnmente posee una estructura compuesta por las siguientes entidades: Título, Objetivo, Contexto, Recursos, Actores, Episodios y Excepciones y el atributo restricción.

Un escenario, identificado por un título, debe satisfacer un objetivo que se alcanza mediante la ejecución de los episodios. Estos representan el curso de acción principal, pero incluyen también variaciones o alternativas posibles. Mientras se ejecutan los episodios puede surgir una excepción, que señala un obstáculo para lograr el objetivo. El contexto se describe detallando una ubicación geográfica, una ubicación temporal y precondiciones. El atributo restricción es usado para caracterizar requisitos no funcionales aplicados a contexto, recursos y episodios (Facultad de Ciencias Exactas, 2007).

- **Storyboards:** Las storyboards en la etapa de especificación de requisitos son usadas para explorar, entender y razonar sobre los requisitos funcionales del sistema, específicamente cómo los usuarios interactuarán con el sistema. Son usadas de conjunto con la técnica escenario. Proporcionan una descripción lógica y conceptual de las funcionalidades del sistema para un escenario determinado, brindándole un mayor realismo a través el uso de medios pictóricos para especificar los requisitos. No constituyen un primer borrador de la interfaz de usuario, sino que están destinadas a representar sólo la interacción del usuario con el sistema (Rational Unified Process, 2007).

1.6.4. Validación de requisitos.

Los requisitos una vez definidos necesitan ser validados. Esta actividad tiene como misión demostrar que la especificación de los requisitos define realmente el sistema que el usuario necesita o el cliente desea, verificando que sus necesidades fueron adecuadamente interpretadas. La validación no debe hacerse sin la participación y presencia de clientes y/o usuarios (Ávila, y otros, 2007).

La comprobación de la consistencia, completitud y corrección de los requisitos constituyen los objetivos de esta etapa de validación además de poder detectar problemas o irregularidades antes de comprometer recursos en su implementación (*Ver Figura 1.13*).

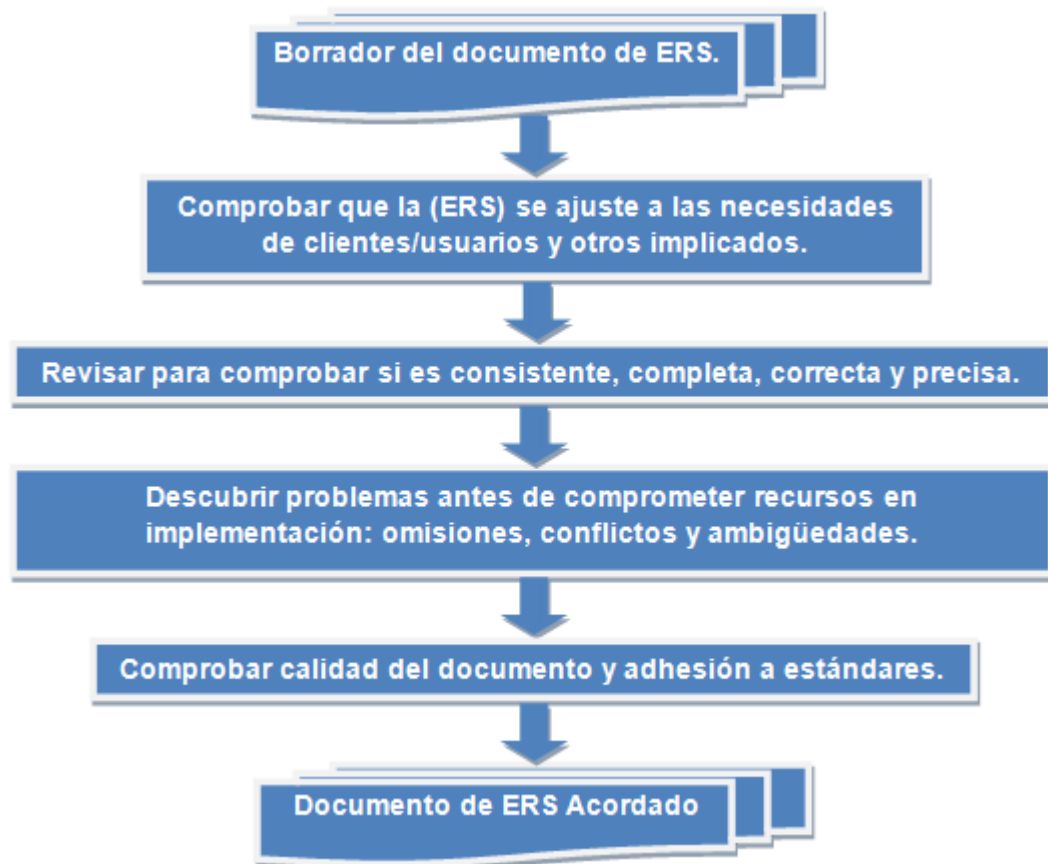


Figura 1.13: Procedimiento para la validación de requisitos.

Técnicas para la validación de los requisitos.

Pocas son las propuestas existentes que ofrecen técnicas para la realización de la validación y muchas de ellas consisten en revisar los modelos obtenidos en la definición de requisitos con el usuario para detectar errores o inconsistencias (Escalona, y otros, 2002). Aún así, se registran algunas técnicas que pueden aplicarse.

- **Revisión (Reviews o Walk-throughs):** Está técnica consiste en la lectura y corrección de la documentación completa o modelado de la definición de requisitos. Con ello solamente se puede validar la correcta interpretación de la información transmitida (Escalona, y otros, 2002).
- **Auditorías:** La revisión de la documentación con esta técnica consiste en un chequeo de los resultados contra una lista de chequeo predefinida o definida a comienzos del proceso, es decir sólo una muestra es revisada (Escalona, y otros, 2002).
- **Prototipos:** Algunas propuestas se basan en obtener de la definición de requisitos prototipos que, sin tener la totalidad de la funcionalidad del sistema, permitan al usuario hacerse una idea

de la estructura de la interfaz del sistema con el usuario. Esta técnica tiene el problema de que el usuario debe entender que lo que está viendo es un prototipo y no el sistema final. Un prototipo puede ser un medio eficaz para entrenar a los clientes en el uso del nuevo software antes de que el producto final esté disponible. Claro, los entrenamientos completos no podrán ser posibles ya que el prototipo probablemente no tendrá las funcionalidades más significativas (Escalona, y otros, 2002).

- **Listas de chequeo (Checklist):** Esta técnica es muy fácil de utilizar. En general es una lista de preguntas que el analista debe usar para evaluar cada requerimiento verificando y marcando los puntos de la lista mientras se lee el documento de requisitos. Cuando se descubren problemas potenciales deben ser anotados, ya sea en los márgenes del documento o en una lista de análisis. Las listas brindan un recordatorio de lo que se debe buscar y reducen la posibilidad de obviar alguna verificación importante (Escalona, y otros, 2002).

1.7. La IR en los Simuladores.

En la encuesta realizada a 10 líderes y/o analistas de proyectos de simulación virtual (*Ver Anexo 1*), el 90% de los encuestados afirma que en los proyectos que han participado han excedido el tiempo y presupuesto inicial (*Ver Figura 1.14*).

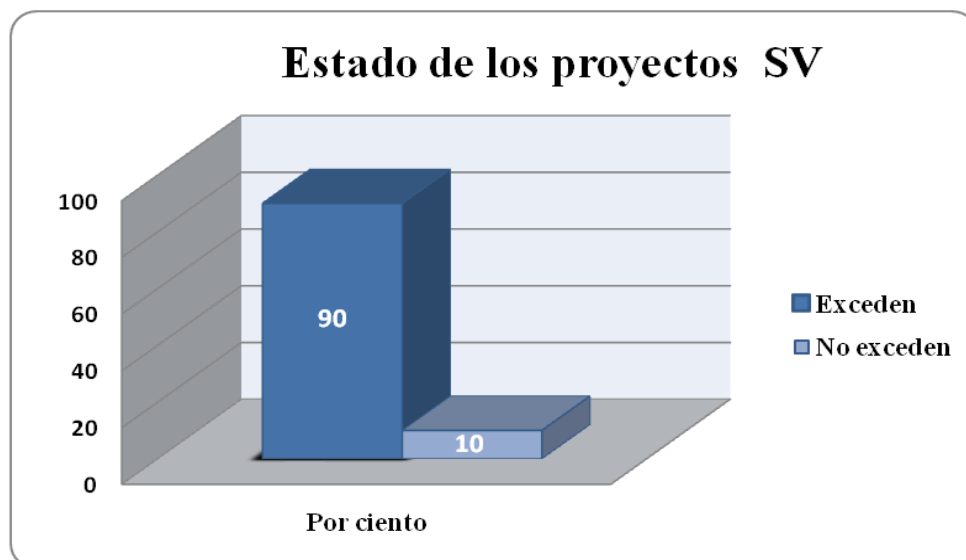


Figura 1.14: Estado de los proyectos de Simulación Virtual.

La prioridad asignada por parte de los encuestados a los requisitos dentro del proceso de desarrollo de software fue alta en un 60%, media en un 30% y baja en un 10% de los casos (*Ver Figura 1.15*).

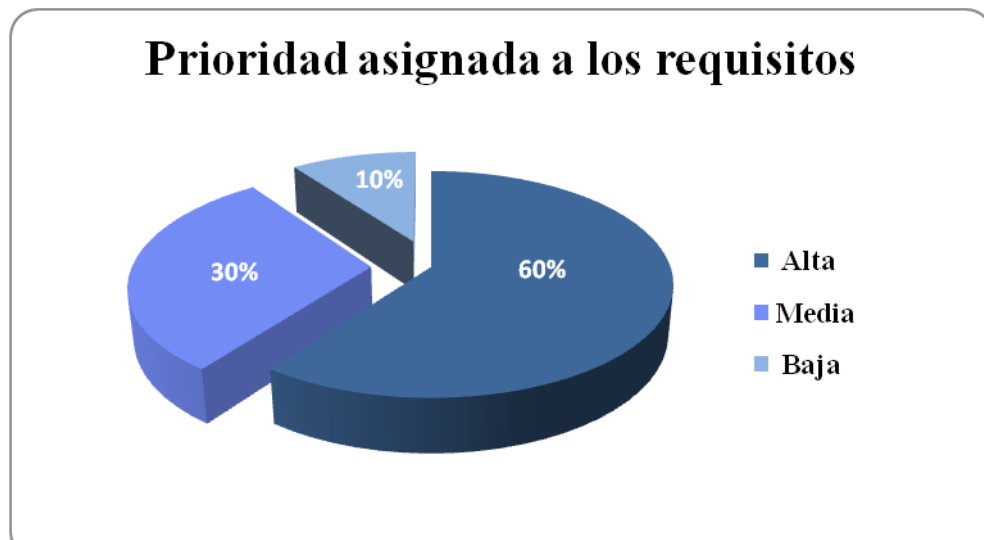


Figura 1.15: Prioridad asignada a los requisitos en proyectos de SV.

Otro de los factores medidos en el diagnóstico fue el nivel de impacto de algunos factores que podrían provocar un exceso en el tiempo y presupuesto en proyectos de simuladores virtuales. (Ver Figura 1.16). Los 5 factores con mayor incidencia son los señalados a continuación y 4 de ellos son provocados por ineficiencia en los procesos de IR.

- Falta de comunicación con usuarios y/o clientes-50%.
- Requisitos y especificaciones incompletas-30%.
- Requisitos y especificaciones cambiantes-30%.
- Estimaciones irreales-40%.
- Falta de recursos-40%.

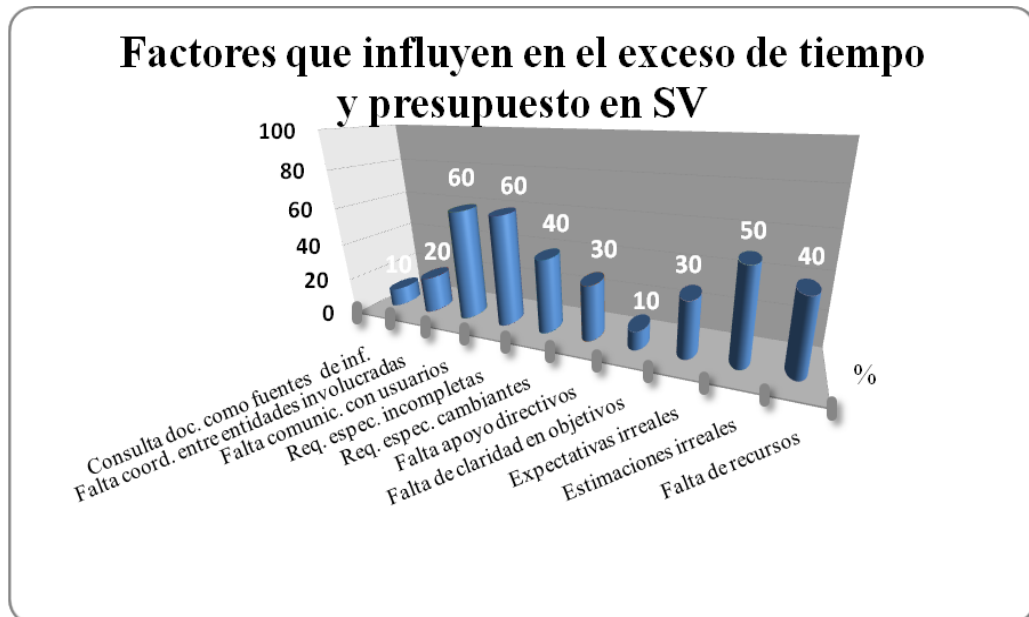


Figura 1.16: Impacto de factores que influyen en el exceso de tiempo y presupuesto en proyectos SV.

Se tiene que el 90% de los encuestados asignan niveles altos y medios a los requisitos en sus proyectos, sin embargo los factores que más inciden en que los proyectos de simulación virtual exceden el tiempo y presupuesto asignados están relacionados con el proceso de desarrollo de requisitos; por lo que se puede concluir que los procedimientos o técnicas utilizadas por los encuestados para el desarrollo de requisitos no son eficientes para este tipo de proyectos.

CAPÍTULO 2: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS.

En este capítulo se abordará la Propuesta de Solución Técnica a desarrollar en la investigación. Para ello se seleccionarán y se desarrollarán las técnicas adecuadas para el desarrollo de requisitos en el Simulador Quirúrgico partiendo de la propuesta tratada anteriormente que incluye las etapas de Elicitación, Análisis, Especificación y Validación de requisitos aunque esta última será tratada con un mayor nivel de detalle en acápites posteriores.

2.1. Estructura del proyecto Simulador Quirúrgico.

El proyecto Simulador Quirúrgico, de manera general, tiene como propósito obtener un prototipo cubano de simulador que aplique la Realidad Virtual al proceso de enseñanza de la cirugía de mínimo acceso, específicamente la cirugía laparoscópica y la endoscopia diagnóstica, basado en la adquisición de imágenes provenientes de pacientes. De esta forma el cirujano podrá no solo entrenarse a partir de una extensa base de ejercicios, sino que además podrá practicar y planificar una intervención quirúrgica antes de realizarla.

El proceso de desarrollo del SimQ se divide básicamente en 4 etapas o fases de desarrollo que generan a su vez 4 productos entregables:

1. Simulador de habilidades básicas, a partir de modelos de imitación de procesos con objetos rígidos.
2. Simulador de habilidades de avanzada a partir de la imitación de procesos, incluyendo endoscopia.
3. Primera versión del simulador a partir de la generación de órganos virtuales.
4. Simulador profesional con órganos adquiridos a partir de scanner (TAC, RM) y retroalimentación al tacto.

Cada una de estos productos constituye una versión funcional a partir de la cual se podrán entrenar las diferentes habilidades a desarrollar en los médicos cirujanos, desde las más básicas hasta el entrenamiento en un simulador profesional en la cuarta y última etapa. Las habilidades podrán ejercitarse a partir de los diferentes ejercicios que comprende cada entregable. Se tiene además la documentación y ayuda necesaria para el entendimiento y posible uso de cada versión funcional (Ver *Figura 2.1*).

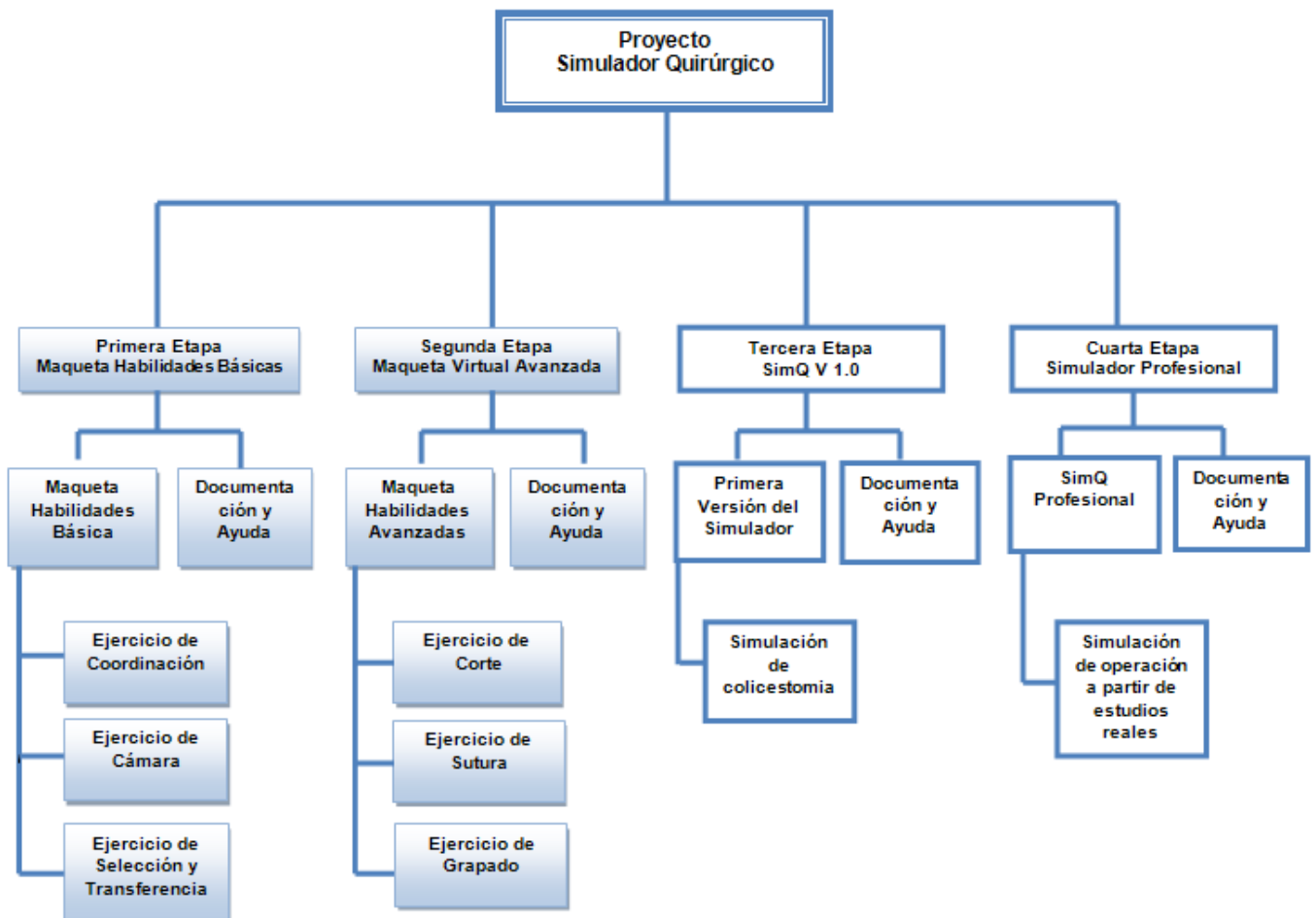


Figura 2.1: Estructura de desglose del proyecto Simulador Quirúrgico.

La propuesta de solución técnica que se presentará en esta investigación estará enmarcada al desarrollo de requisitos de todo el simulador y se aplicará a sus dos primeras etapas o fases de desarrollo: Maqueta de habilidades básicas y Maqueta virtual avanzada debido a la correspondencia del período actual con el cronograma de ejecución del proyecto. La valoración de los desarrolladores del proyecto acerca de los resultados obtenidos con la aplicación de la propuesta en los restantes flujos de trabajo del ciclo de vida del proyecto se podrá realizar solamente a partir de la experiencia de la primera etapa de desarrollo.

2.2. Propuesta de Elicitación de Requisitos para el SimQ.

En este apartado se seleccionan las técnicas a utilizar para la obtención de requisitos en el SimQ teniendo en cuenta el estudio de la teoría analizada en epígrafes preliminares en cuanto a la Elicitación de Requisitos. Para lograr una comprensión adecuada de los requisitos será preciso utilizar varias

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

técnicas a partir de las cuales se podrán adquirir datos suficientes, relevantes y apropiados para definir un conjunto estable de requisitos y llegar a conocer la forma en que el médico cirujano realiza las tareas de entrenamiento a las que el SimQ dará soporte, las metas asociadas a dicho entrenamiento, así como el contexto en el que se realiza.

A continuación se detallan las técnicas seleccionadas para la obtención de los requisitos de SimQ:

La entrevista: Las personas entrevistadas serán básicamente clientes beneficiados con el SimQ, aunque también pueden formar parte de este proceso otros profesionales en el área de la cirugía endoscópica que puedan aportar información y precisar detalles prácticos a tener en cuenta durante el entrenamiento quirúrgico que servirán como base para definir las futuras funcionalidades del simulador. La entrevista será realizada por el analista del proyecto (entrevistador) aunque puede estar acompañado también por el líder del proyecto.

EL diálogo tendrá una duración de solo una hora evitando, de esta forma, agotamiento en los entrevistados o importunar por mucho tiempo su horario de trabajo. El analista previo a la realización de la entrevista debe estudiar categorías y conceptos relacionados con la Cirugía de Mínimo Acceso y el entrenamiento realizado por los médicos de esta especialidad con el fin de entender las necesidades del entrevistado y lograr una mejor comunicación durante el proceso.

La entrevista debe ser confeccionada de manera que abarque todo el contexto del entrenamiento que realizan los médicos cirujanos. Para ello el analista, conjuntamente con el líder del proyecto, definirá previamente algunas interrogantes o ideas que propicien un marco de conversación agradable y a la vez sirvan como guía. El objetivo de las preguntas no consiste en obtener una respuesta cerrada por parte del entrevistado, sino lograr un mayor volumen de información que permita un entendimiento de las actividades involucradas en el proceso de la cirugía de mínimo acceso, y a la vez a evitar la sensación de interrogatorio en el entrevistado proporcionando fluidez a la conversación.

Entre estas preguntas podemos encontrar: ¿Cómo se realiza el entrenamiento de los médicos cirujanos para una Cirugía de Mínimo Acceso?, ¿Qué tipo de instrumental quirúrgico se utiliza en una Cirugía de Mínimo Acceso?, ¿Qué habilidades se desean entrenar y medir en los estudiantes de la especialidad de cirugía endoscópica con el empleo del SimQ?, por solo citar algunas (*Ver Anexo 2*). Se tendrá especial cuidado en el empleo de términos técnicos propios de la formación de los analistas que el entrevistado no conozca y que puedan afectar la comunicación.

Concluida la entrevista el analista se procede a organizar la información recopilada con el fin de utilizarla en función de definir más adelante la lista de requisitos a implementar en el SimQ.

Aprendiz: Como se planteaba anteriormente en la confección de un SimQ se manifiesta una estrecha relación de dos áreas tan diferentes como la informática y la medicina, evidentemente los informáticos, que constituyen la fuerza de trabajo calificada para la confección del sistema que automatizará el entrenamiento médico, no cuentan con el conocimiento necesario para asimilar y entender los detalles que comprenden el proceso en cuestión. La situación se torna más difícil si a esto se le añade la complejidad y dimensión del entrenamiento a simular. Es por ello que se propone el uso de la técnica del aprendiz de manera que el analista y/o líder del proyecto pueda participar conjuntamente con el médico cirujano durante su proceso de entrenamiento con el objetivo de observar directamente los detalles más mínimos e imprescindibles a tener en cuenta ya que en ocasiones se torna engorroso un entendimiento basado en palabras, más aún si se tiene en cuenta que la información a modelar percibe un nivel de detalle y precisión muy elevado.

Tormenta de ideas: Con el objetivo de acumular ideas, información, criterios y puntos de vista que permitan analizar y precisar las funcionalidades exactas que comprenderán el producto del SimQ que servirán a los médicos cirujanos en el entrenamiento dentro del campo de la CMA. Para concretar la puesta en marcha de esta técnica se definirán sesiones de tormentas de ideas con un número máximo de 6 participantes: el líder de proyecto, el analista (moderador de la sesión), el especialista de hardware y el cliente (médico cirujano con conocimientos de realidad virtual), además de uno o dos médicos más con conocimiento y experiencia en la CMA de manera que cada uno pueda mostrar sus ideas de forma libre y espontánea.

Para iniciar la sesión se toman como punto de partida los objetivos del SimQ a partir de los cuales los participantes en la reunión formularán sus ideas. Es preciso recoger todas las ideas expuestas por los participantes fomentando las más avanzadas y teniendo siempre muy presente que no se deben realizar críticas. A partir de las ideas iniciales propuestas por los presentes en la reunión se van generando nuevas rondas de ideas o ideas derivadas con la motivación de producir el mayor número posible de ellas. Al finalizar se procederá a organizar las ideas propuestas durante la reunión, se descartan las ideas excesivamente avanzadas y se priorizan las restantes clasificándolas en tres grupos: las absolutamente esenciales, las que estarían bien pero no son lo suficientemente esenciales o importantes y las que podrían ser apropiadas para una próxima versión del SimQ.

2.3. Propuesta de Análisis de Requisitos para el SimQ.

En esta etapa se toma como punto de partida la información extraída por el analista mediante la aplicación de las técnicas para la obtención de los requisitos propuestas en el epígrafe anterior. Los

requisitos obtenidos a partir de la observación de una sesión de entrenamiento de un médico cirujano y mediante el análisis de la entrevista realizada a los clientes y las ideas generadas en la sesión de tormenta de ideas son transformados en condiciones apropiadas para ser tratadas en el diseño del sistema. Estos requisitos se formalizan por escrito, estableciendo los servicios que el SimQ debe proporcionar y las restricciones bajo las cuales debe operar, así como las condiciones que determinan qué debe hacer, o sea se clasifican en funcionales y no funcionales. Es importante verificar que los requisitos cumplan con las características definidas en el capítulo anterior a fin de evitar cualquier tipo de contradicción.

2.3.1. Requisitos Funcionales del SimQ.

Primera Etapa: Maqueta de habilidades básicas.

Para esta primera etapa de desarrollo del proyecto SimQ el sistema debe permitir:

1. R-1. Autenticar Usuario.

2. Gestionar Usuarios.

R-2.1. Insertar datos de un nuevo usuario.

R-2.2. Modificar los datos de un usuario existente en la base de datos.

R-2.3. Eliminar los datos de un usuario existente en la base de datos.

3. Inicializar Ejercicio.

R-3.1. Crear cámara.

R-3.2. Posicionar cámara.

R-3.3. Crear luces.

R-3.4. Posicionar luces.

R-3.5. Cargar el modelo del intestino.

R-3.6. Cargar el modelo del torso.

R-3.7. Cargar la pinza.

R-3.8. Posicionar pinza.

4. Gestionar Nivel de Entrenamiento primera etapa.

R-4.1. Seleccionar nivel (básico, medio o avanzado).

R-4.2. Asignar cantidad de objetos según el nivel de entrenamiento seleccionado (nivel básico: 5 objetos, nivel intermedio: 10 objetos, nivel experto: 15 objetos).

R-4.3. Cargar los objetos.

R-4.4. Posicionar objetos.

5. Resetear Ejercicio.

R-5.1. Decrementar la cantidad de resets permitidos en el ejercicio.

R-5.2. Posicionar los elementos del ejercicio en sus posiciones iniciales.

R-5.3. Mostrar cantidad de resets permitidos (Máximo de resets permitidos igual a 3).

Ejercicio de la Cámara:

6. Interactuar con la cámara.

R-6.1. Mover cámara según movimiento del Mouse y/o teclado.

R-6.2. Mover luz junto con la cámara.

7. Focalizar objeto con la cámara.

R-7.1. Animar el objeto a focalizar con una luz parpadeante mientras no se encuentre focalizado.

R-7.2. Detectar objeto focalizado.

R-7.3. Animar textura del objeto al ser focalizado.

R-7.4. Activar reloj de conteo regresivo cuando el objeto se encuentre focalizado (Tiempo máximo 10 segundos).

R-7.5. Decrementar cantidad de objetos que aun no han sido focalizados cuando el conteo del reloj regresivo llegue a cero.

R-7.6. Devolver al objeto que se encuentra focalizado su textura original cuando el conteo del reloj regresivo llegue a cero.

8. Detectar Colisión Cámara.

R-8.1. Detectar colisión cámara-objeto.

R-8.2. Detectar colisión cámara-intestino.

R-8.3. Hacer reaccionar la cámara al choque (aplicar fuerza física).

9. Mostrar estadísticas Cámara.

R-9.1. Mostrar cantidad de objetos que aún no han sido focalizados.

R-9.2. Mostrar estadísticas de errores de focalización de la cámara (Cantidad de objetos que no fueron focalizados correctamente).

R-9.3. Mostrar conteo de reloj regresivo.

R-9.4. Mostrar estadísticas de errores por concepto de colisión (Cantidad de colisiones cámara-objeto y cámara-intestino).

R-9.5. Mostrar estadísticas del tiempo de indecisión en el movimiento de la cámara (Tiempo que la cámara permanece estática, sin moverse).

Ejercicio Coordinación/Selección:

10. Interactuar con pinzas.

R-10.1. Seleccionar pinza a activar.

R-10.2. Mover pinzas según dirección del Mouse.

11. Detectar colisión.

R-11.1. Detectar colisión pinza-torso.

R-11.2. Detectar colisión pinza-pinza.

R-11.3. Hacer reaccionar la pinza activada al choque (aplicar fuerza física).

R-11.4. Mostrar estadísticas de errores por concepto de colisión (Cantidad de colisiones pinza-pinza, pinza-torso).

R-11.5. Detectar colisión pinza-objeto.

12. R-12. Mostrar estadísticas del tiempo de indecisión (Tiempo que la pinza permanece estática, sin moverse).

Ejercicio de Coordinación:

13. Seleccionar Objetos.

R-13.1. Simular explosión de un objeto al colisionar con la pinza incorrecta (El Gancho (Hook) selecciona los objetos señalizados con una "H" y el Pinza (Scissor) los señalizados con una "S").

R-13.2. Incrementar estadísticas de errores de selección al colisionar un objeto con la pinza incorrecta.

R-13.3. Desaparecer objeto de la escena al colisionar con la pinza correcta (El Gancho (Hook) selecciona los objetos señalizados con una "H" y el Pinza (Scissor) los señalizados con una "S").

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

R-13.4. Incrementar estadísticas de acierto al colisionar un objeto con la pinza correcta.

R-13.5. Decrementar cantidad de objetos que aun no han sido seleccionados.

14. Mostrar estadísticas Coordinación.

R-14.1. Mostrar estadísticas de errores por concepto de selección incorrecta (cantidad de colisiones pinza-objeto incorrectas).

R-14.2. Mostrar estadística de aciertos (cantidad de colisiones pinza-objeto correctas).

R-14.3. Mostrar cantidad de objetos que aun no han sido seleccionados.

Ejercicio de Selección:

15. Agarrar y transferir Objetos.

R-15.1. Abrir y cerrar las muelas de las pinzas (Angulo de amplitud máximo 60°).

R-15.2. Agarrar el objeto con las muelas de la pinza (Angulo de amplitud necesario para agarrar el objeto 45°).

R-15.3. Mover el objeto junto a la pinza activada.

R-15.4. Hacer reaccionar el objeto al choque (Cuando el ángulo de amplitud entre las muelas de la pinza es menor de 45°) (aplicar fuerza física).

R-15.5. Incrementar estadísticas de errores por concepto de selección incorrecta (La pinza azul manipula los objetos azules y la pinza roja los objetos rojos).

R-15.6. Depositar el objeto.

R-15.7. Incrementar estadísticas de aciertos (Objeto depositado por la pinza correcta y dentro del área de depósito).

R-15.8. Decrementar cantidad de objetos.

R-15.9. Incrementar estadísticas de cantidad de objetos caídos (Objeto depositado fuera del área de depósito).

16. Mostrar estadísticas Selección.

R-16.1. Mostrar estadísticas de cantidad de aciertos (Cantidad de objetos que fueron agarrados y depositados correctamente).

R-16.2. Mostrar estadísticas de errores de por concepto de selección incorrecta (Cantidad de objetos agarrados con la pinza incorrecta).

R-16.3. Mostrar estadísticas de cantidad de objetos caídos (Cantidad de objetos que fueron depositados incorrectamente).

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

R-16.4. Mostrar cantidad de objetos que aun no han sido transferidos.

R-16.5. Mostrar estadísticas de errores por concepto de colisión (Cantidad de colisiones pinza-objeto).

Segunda Etapa: Maqueta de habilidades avanzadas.

17. Inicializar Ejercicio Corte/Sutura.

R-17.1. Cargar el modelo de la escena del corte.

R-17.2. Cargar el modelo del Intestino.

R-17.3. Cargar porta-aguja.

R-17.4. Posicionar porta-aguja.

R-17.5. Cargar aguja e hilo.

R-17.6. Posicionar aguja e hilo.

R-17.7. Cargar la tijera.

R-17.8. Posicionar la tijera.

18. Inicializar Ejercicio de Grapado.

R-18.1. Cargar el modelo de la escena de grapado.

R-18.2. Cargar grapa.

R-18.3. Cargar grapadora.

R-18.1. Posicionar grapadora.

19. Gestionar Nivel de Entrenamiento segunda etapa.

R-19.1. Asignar trayectoria a la que debe ajustarse el corte (nivel básico: trayectoria recta, nivel intermedio: trayectoria curva, nivel experto: trayectoria circular).

R-18.2. Asignar técnicas de sutura a realizar según el nivel de entrenamiento seleccionado (nivel básico: sutura simple, nivel intermedio: sutura continua y nivel experto: sutura discontinua).

R-18.3. Asignar cantidad de grapas a colocar (nivel básico: 5 objetos, nivel intermedio: 10 objetos, nivel experto: 15 objetos).

Ejercicio de Corte:

20. Detectar Colisión Corte.

- R-20.1. Detectar colisión tijera-pinza.
- R-20.2. Hacer reaccionar la tijera al choque (aplicar fuerza física).
- R-20.3. Detectar colisión tijera-escena.

21. Gestionar Ejercicio Corte.

- R-21.1. Activar la tijera.
- R-21.2. Abrir y cerrar la tijera.
- R-21.3. Mover la tijera según los movimientos del mouse.
- R-21.4. Realizar corte.
- R-21.5. Deformar la escena a medida que se realice el corte.

22. Mostrar Estadísticas Corte.

- R-22.1. Mostrar estadísticas del tiempo de indecisión en el movimiento de la tijera (Tiempo que la tijera permanece estática, sin moverse).
- R-22.2. Mostrar estadísticas de errores por concepto de corte incorrecto (El corte realizado no se ajusta a la trayectoria definida para el mismo).
- R-22.3. Mostrar estadísticas de aciertos (Ajuste del corte realizado a la trayectoria definida para la realización del mismo).
- R-22.4. Mostrar estadísticas de errores por concepto de colisión (Cantidad de colisiones tijera-pinza).

Ejercicio de Sutura:

23. Detectar Colisión Sutura.

- R-23.1. Detectar colisión porta-aguja-pinza.
- R-23.2. Detectar colisión tijera-pinza.
- R-23.3. Hacer reaccionar la herramienta activada al choque (aplicar fuerza física).
- R-23.4. Detectar colisión aguja-intestino.
- R-23.5. Detectar colisión tijera-hilo.

24. Gestionar Ejercicio Sutura.

R-24.1. Activar el porta-agujas.

R-24.2. Mover el porta-agujas según dirección del Mouse.

R-24.3. Realizar sutura.

R-24.4. Deformar el intestino a medida que se realice la sutura.

R-24.5. Incrementar estadísticas de aciertos (Sumar un 1 punto cada vez que la aguja entre o salga por el punto correcto y 2 puntos cuando el tipo de nudo realizado esté en correspondencia con la técnica de sutura asignada al nivel en cuestión).

R-24.6. Incrementar cantidad de errores cometidos (Sumar un 1 punto cada vez que la aguja entre o salga fuera del punto correcto y 1 punto cuando el tipo de nudo realizado no esté en correspondencia con la técnica de sutura asignada al nivel en cuestión).

R-24.7. Cortar el hilo con la tijera.

25. Mostar Estadísticas Sutura.

R-25.1. Mostrar estadísticas del tiempo de indecisión en el movimiento del porta-agujas (Tiempo que el porta-agujas permanece estático, sin moverse).

R-25.2. Mostrar cantidad de errores cometidos (los puntos de entrada y salida de la aguja no se ajustan a los definidos en el ejercicio y del tipo de nudo realizado no corresponde al asignado por el nivel de entrenamiento en cuestión).

R-25.3. Mostrar estadística de aciertos (los puntos de entrada y salida de la aguja se ajustan a los definidos en el ejercicio y del tipo de nudo realizado corresponde al asignado por el nivel de entrenamiento en cuestión).

R-25.4. Mostrar estadísticas de errores por concepto de colisión (Cantidad de colisiones porta-agujas-pinza, tijera-pinza).

Ejercicio de Grapado:

26. Detectar colisión Grapado.

R-26.1. Detectar colisión grapadora-escena.

R-26.2. Detectar colisión grapadora-pinza.

27. Gestionar Ejercicio Grapado.

R-27.1. Mover grapadora según dirección del Mouse.

R-27.2. Posicionar grapa.

R-27.3. Deformar el objeto a medida que se realice el grapado.

R-27.4. Incrementar estadísticas de aciertos (Sumar 1 punto por cada grapa posicionada correctamente según los putos definidos para su colocación)

R-27.5. Incrementar cantidad de errores cometidos (Sumar 1 punto por cada grapa posicionada fuera de los putos definidos para su colocación).

R-27.6. Mostrar cantidad de grapas a colocar según el nivel de complejidad en cuestión.

28. Mostrar Estadísticas Grapado.

R-28.1. Mostrar estadísticas del tiempo de indecisión en el movimiento de la grapadora (Tiempo que la grapadora permanece estática, sin moverse).

R-28.2. Mostrar cantidad de errores cometidos (no correspondencia de la posición en que fueron ubicadas las grapas con los puntos definidos para su colocación).

R-28.3. Mostrar estadística de aciertos (cantidad de grapas colocadas en los puntos definidos).

29. Emitir Evaluación.

R-29.1. Guardar la evaluación obtenida del ejercicio realizado.

R-29.2. Mostrar la evaluación obtenida del ejercicio realizado por el médico cirujano.

30. R-30. Mostrar la evaluación obtenida por cualquier médico en cualquier ejercicio al Profesor.

2.3.2. Requisitos No Funcionales del SimQ.

- **Usabilidad:** Los futuros usuarios del sistema serán médicos cirujanos que se entrenarán con el SimQ antes de enfrentarse a una CMA real. Este producto debe poseer una interfaz del sistema en la cual aparezcan las herramientas a utilizar por el médico de modo que sea fácil de aprender el funcionamiento de la aplicación.
- **Rendimiento:** Como toda aplicación de tiempo real, debe tener alta velocidad de procesamiento o cálculo, tiempo de respuesta y de recuperación, y disponibilidad. La cantidad de frames por segundo (**FPS**) debe ser superior a 30 e inferior a 60.
- **Soporte:** Una vez instalado el producto se le dará mantenimiento por parte de un equipo de especialista de Hardware. El producto contará con un manual que explicará el funcionamiento del mismo.
- **Hardware:** Compatibilidad con tarjetas gráficas de la familia NVIDIA (Quadro FX 500/FX 600) Procesador superior a 1GHz de velocidad y RAM superior a 512 MB.

- **Portable:** El producto debe ser producido de modo que funcione tanto en Windows como en Linux.
- **Restricciones en el diseño y la implementación:** Debe utilizar transparentemente el *engine* gráfico STK en su versión 2.4. Será implementado con la herramienta Visual Studio 2003, con la utilización del lenguaje C/C++, regido por la teoría de Programación Orientada a Objetos. Para la implementación del módulo de Base de Datos se va utilizar el *Firebird* versión 5.0.

2.4. Propuesta de Especificación de Requisitos para el SimQ.

Hasta el momento en la mayoría de los proyectos de simulación desarrollados en el país, la etapa de Especificación de Requisitos se ha realizado mediante el conocido método de los casos de uso, incluso en algunos de ellos ni siquiera se ha tenido en cuenta esta técnica y se han limitado a una simple especificación por escrito de los requisitos. Esto provoca que en fases posteriores del desarrollo del software se detecten deficiencias derivadas de una incorrecta definición de los mismos. Unido a esto se tiene el hecho de que los simuladores en general, y por ende el SimQ, presenta particularidades que los diferencian del resto de los sistemas software. Se caracterizan por una amplia interactividad con los usuarios y un procesamiento especial de imagen, involucrando sentidos como la vista, el tacto y el oído, así como la necesidad de mostrar un mayor realismo en las escenas y de obtener respuestas rápidas y eficientes por parte del sistema, especificidades que se hacen muy difíciles de representar mediante una simple descripción textual del proceso.

Con el objetivo de dar solución a estos problemas se propone el uso combinado de los **Escenarios** y los **Casos de Uso** como técnica para la especificación de los requisitos del SimQ. La técnica casos de uso se aplicará fundamentalmente a aquellos requisitos que no necesitan de una representación gráfica para lograr una comprensión detallada por parte de los clientes y desarrolladores, sino que basta con una minuciosa lectura del proceso que se describe a partir de la secuencia de pasos desarrollados en las secciones del flujo normal y alterno de eventos. Una propuesta a utilizar esta técnica es en la especificación de los requisitos que responden a funcionalidades a implementar en el Módulo de Base de Datos. Para la descripción textual de los Casos de Uso se empleará la plantilla estándar definida por RUP en (Rational Unified Process, 2003) (*Ver Anexo 3*).

La técnica de escenarios permite hacer una representación de los requisitos mediante videos y storyboard, alternativa esta que se ajusta a las necesidades descritas, pues su uso permite mostrar de una forma más precisa algunos requisitos que lleven implícito gran interactividad con el usuario y que exigen una representación gráfica para lograr un mejor entendimiento de los mismos.

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

Para la conformación de los escenarios se hará uso de una plantilla definida en este trabajo (Ver Anexo 4), tomando en cuenta los elementos más importantes presentados en (Hadad, y otros, 1999) donde se reserva un espacio que permite mostrar los videos y/o storyboard que describan los requisitos involucrados dentro de cada escenario, con el propósito de visualizar las funcionalidades del sistema con la mayor eficiencia posible.

A partir del estudio de los usuarios potenciales del SimQ se identifican los siguientes perfiles de usuarios:

- **Administrador del Sistema:** Es el encargado de gestionar la información referente a los usuarios, asignando los permisos correspondientes según los roles definidos.
- **Médico Cirujano:** Utilizará el sistema para entrenar y crear habilidades a través de los ejercicios.
- **Profesor:** Usuario que también podrá entrenarse al igual que el usuario Médico Cirujano, pero que además podrá conocer las evaluaciones obtenidas por estos médicos cirujanos en cada una de sus sesiones de entrenamiento.

2.4.1. Especificación de los requisitos correspondientes a la primera y segunda etapa del SimQ.

Primeramente se presentarán las especificaciones a partir de la técnica de Casos de Uso siguiendo la distribución de los requisitos funcionales definidos en el análisis de requisitos (Ver Tabla 2.1):

Tabla 2.1: Casos de Uso definidos para la primera y segunda etapa de desarrollo del SimQ.

Caso de Uso	Requisitos Funcionales	Etapas de Desarrollo
Autenticar Usuario	R-1.	E1
Gestionar Usuario	R-2.1, R-2.2, R-2.3.	E1
Iniciar Ejercicio	R-3.1, R-3.2, R-3.3, R-3.4, R-3.5, R-3.6, R-3.7, R-3.8, R-4.1, R-4.2, R-4.3, R-4.4, R-17.1, R-17.2, R-17.3, R-17.4, R-17.5, R-17.6, R-17.7, R-17.8, R-18.1, R-18.2, R-18.3, R-18.4, R-19.1,	E1 y E2

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

	R-19.2, R-19.3.	
Resetear Ejercicio	R-5.1, R-5.2, R-5.3.	E1 y E2
Emitir Evaluación	R-29.1, R-29.2.	E2
Gestionar Evaluación	R-30.	E2

Tabla 2.2: Descripción textual del caso de uso Autenticar Usuarios.

Caso de Uso:	Autenticar Usuarios.	
Actores:	Usuario del Sistema (inicia).	
Resumen:	El caso de uso inicia cuando un Usuario del Sistema decide autenticarse para acceder al sistema.	
Precondiciones:	El Usuario del Sistema debe haber sido registrado previamente como usuario del Simulador.	
Referencias	R-1.	
Prioridad	Crítico.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El Usuario del Sistema inserta los datos solicitados por el sistema: usuario y contraseña.	<p>1.1. El sistema verifica la validez de los caracteres de entrada para el usuario y la contraseña insertados por el Usuario del Sistema.</p> <p>1.2. El sistema encripta la contraseña insertada por el Usuario del Sistema.</p> <p>1.3. El sistema verifica el usuario insertado y la contraseña encriptada</p>	

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

	<p>con los almacenados en la Base de Datos.</p> <p>1.4. Se le asignan los permisos correspondientes al rol en cuestión. Finalizando así el caso de uso.</p>
Flujo Alterno	
	<p>1.1. Si el sistema comprueba la existencia de caracteres no válidos, muestra un mensaje “Caracteres de entrada no válidos”</p> <p>1.3. Si el sistema comprueba que los datos insertados no son validos, muestra un mensaje avisando que: “Su usuario o contraseña son incorrectos” y brinda la posibilidad al Médico Cirujano de insertar nuevamente los datos de autenticación. Si después de varios intentos el sistema comprueba que no existe el usuario en cuestión, muestra un mensaje avisando que: “El usuario no existe, debe registrarse como usuario del sistema”.</p>
Poscondiciones	Se ha autenticado un usuario en el sistema y se habilitan los permisos según sus privilegios.

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

Tabla 2.3: Descripción textual del caso de uso Gestionar Usuarios del Sistema.

Caso de Uso:	Gestionar Usuarios del Sistema	
Actores:	Administrador del Sistema (Inicia)	
Resumen:	El caso de uso se inicia cuando el Administrador del Sistema decide registrar, modificar o eliminar datos acerca de un usuario.	
Precondiciones:	Administrador del sistema previamente autenticado.	
Referencias	R-2.1, R-2.2, R-2.3.	
Prioridad	Crítico	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El Administrador del Sistema decide registrar, modificar o eliminar los datos de un usuario.	<p>1.1. El sistema ejecuta algunas de las siguientes acciones:</p> <ul style="list-style-type: none"> a) Si decide registrar un usuario, ir a la sección "Registrar usuario" b) Si decide modificar los datos de un usuario, ir a la sección "Modificar usuario". c) Si decide eliminar un usuario, ir a la sección "Eliminar usuario". 	
Sección "Registrar Usuario"		
Flujo Normal		
Acción del Actor	Respuesta del Sistema	
1. El Administrador del Sistema inserta los datos necesarios para registrar un nuevo usuario en el sistema (Nombre, Apellidos, Usuario, Contraseña, Dirección Particular,	<p>1.1. El sistema verifica que todos los campos estén llenos.</p> <p>1.2. El sistema verifica la validez de los</p>	

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

Teléfono, Centro de Trabajo, Años de Experiencia), además del rol que jugará este en el sistema.

caracteres de entrada.

- 1.3. El sistema verifica que esta persona no haya sido registrada anteriormente en el sistema.
- 1.4. Almacena los datos del usuario en la Base de Datos.
- 1.5. Se muestra un mensaje informándosele al Administrador del Sistema que ha sido efectuado correctamente el registro del usuario. Finalizando así el caso de uso.

Flujo Alternativo

- 1.1. Si falta algún campo por llenar, se emite un mensaje informando que todos los campos deben de estar llenos.
- 1.2. Si el sistema comprueba la existencia de caracteres no válidos, muestra un mensaje "Caracteres de entrada no válidos"
- 1.3. Si el usuario existe, se emite un mensaje informando la existencia del mismo. Finalizando así el caso de uso.

Sección "Modificar Usuario"

Flujo Normal

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

Acción del Actor	Respuesta del Sistema
1. El Administrador del Sistema selecciona el usuario para el cual desea modificar datos.	1.1. El sistema muestra una interfaz que brinda la posibilidad de modificar los datos existentes.
2. El Administrador del Sistema realiza las actualizaciones deseadas.	2.1. El sistema verifica que todos los campos estén llenos. 2.2. Se actualiza en la Base de Datos la información modificada y se emite un mensaje informando que la modificación se realizó satisfactoriamente. Finalizando así el caso de uso.
Flujo Alterno	
	2.1. Si falta algún campo por llenar, se emite un mensaje informando que todos los campos deben de estar llenos.
Sección “Eliminar Usuario”	
Flujo Normal	
Acción del Actor	Respuesta del Sistema
1. El Administrador del Sistema selecciona el usuario que desea eliminar.	1.1. El sistema solicita confirmación para eliminar el usuario.
2. El Administrador del Sistema confirma que desea eliminar este usuario.	2.1. El sistema elimina los datos del usuario seleccionado de la Base de Datos y emite un mensaje

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

	informando que la operación se ha realizado de forma satisfactoria. Finalizando así el caso de uso.
Poscondiciones	<p>Sección “Registrar Usuario”: Se ha registrado un nuevo usuario en el sistema y se le han habilitado las funcionalidades y permisos correspondientes según sus privilegios.</p> <p>Sección “Modificar Usuario”: Se han modificado los datos de un usuario en la Base de Datos.</p> <p>Sección “Eliminar Usuario”: Se han eliminado correctamente los datos de un usuario de la Base de Datos.</p>

Tabla 2.4: Descripción textual del caso de uso Iniciar Ejercicio.

Caso de Uso:	Iniciar Ejercicio
Actores:	Médico Cirujano (inicia).
Resumen:	El caso de uso inicia cuando el Médico Cirujano decide iniciar un ejercicio de entrenamiento: Cámara, Coordinación, Selección, Corte, Sutura o Grapado.
Precondiciones:	Médico Cirujano previamente autenticado.
Referencias	R-3.1, R-3.2, R-3.3, R-3.4, R-3.5, R-3.6, R-3.7, R-3.8, R-4.1, R-4.2, R-4.3, R-4.4, R-17.1, R-17.2, R-17.3, R-17.4, R-17.5, R-17.6, R-17.7, R-17.8 R-18.1, R-18.2, R-18.3, R-18.4, R-19.1, R-19.2, R-19.3.
Prioridad	Crítico
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Médico Cirujano decide entrenarse en uno de los ejercicios: Cámara, Coordinación,	1.1. Si el Médico Cirujano decide entrenarse el ejercicio de la cámara,

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

<p>Selección, Corte, Sutura o Grapado.</p>	<p>ir a sección “Iniciar Cámara”.</p> <p>1.2. Si el Médico Cirujano decide entrenarse el ejercicio de coordinación o el ejercicio de selección, ir a sección “Iniciar Coordinación/Selección”.</p> <p>1.3. Si el Médico Cirujano decide entrenarse el ejercicio de corte, ir a sección “Iniciar Corte”.</p> <p>1.4. Si el Médico Cirujano decide entrenarse en el ejercicio de sutura, ir a sección “Iniciar Sutura”.</p> <p>1.5. Si el Médico Cirujano decide entrenarse el ejercicio de grapado, ir a sección “Iniciar Grapado”.</p>
Sección “Iniciar Cámara”	
Flujo Normal	
Acción del Actor	Respuesta del Sistema
<p>1. El Médico Cirujano selecciona el ejercicio de la cámara.</p>	<p>1.1. El sistema carga el modelo del intestino.</p> <p>1.2. El sistema crea y posiciona la cámara.</p> <p>1.3. El sistema crea y posiciona las luces.</p>
<p>2. El Médico Cirujano selecciona el nivel en el que desea entrenarse: básico, medio,</p>	<p>2.1. El sistema adiciona los objetos a la escena en correspondencia con el</p>

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

avanzado.	nivel de entrenamiento elegido por el Médico Cirujano.
Sección “Iniciar Coordinación/Selección”	
Flujo Normal	
Acción del Actor	Respuesta del Sistema
1. El Médico Cirujano selecciona el ejercicio de coordinación o el ejercicio de selección.	<p>1.1. El sistema carga el modelo del torso.</p> <p>1.2. El sistema crea y posiciona la cámara.</p> <p>1.3. El sistema crea y posiciona las luces.</p> <p>1.4. El sistema crea y posiciona las pinzas.</p>
2. El Médico Cirujano selecciona el nivel en el que desea entrenarse: básico, medio, avanzado.	2.1. El sistema adiciona los objetos a la escena en correspondencia con el nivel de entrenamiento elegido por el Médico Cirujano.
Sección “Iniciar Corte”	
Flujo Normal	
Acción del Actor	Respuesta del Sistema
1. El Médico Cirujano selecciona el ejercicio de corte.	<p>1.1. El sistema carga el modelo de la escena del corte</p> <p>1.2. El sistema crea y posiciona la cámara.</p>

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

	<p>1.3. El sistema crea y posiciona las luces.</p> <p>1.4. El sistema crea y posiciona las pinzas.</p> <p>1.5. El sistema crea y posiciona la tijera.</p>
<p>2. El Médico Cirujano selecciona el nivel en el que desea entrenarse: básico, medio, avanzado.</p>	<p>2.1. El sistema asigna la trayectoria a la que debe ajustarse el corte según el nivel de entrenamiento elegido por el Médico Cirujano.</p>
Sección “Iniciar Sutura”	
Flujo Normal	
Acción del Actor	Respuesta del Sistema
<p>1. El Médico Cirujano selecciona el ejercicio de sutura.</p>	<p>1.1. El sistema carga el modelo del Intestino.</p> <p>1.2. El sistema crea y posiciona la cámara.</p> <p>1.3. El sistema crea y posiciona las luces.</p> <p>1.4. El sistema crea y posiciona las pinzas.</p> <p>1.5. El sistema crea y posiciona el porta-agujas.</p> <p>1.6. El sistema crea y posiciona la aguja.</p> <p>1.7. El sistema crea y posiciona el hilo.</p>

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

<p>2. El Médico Cirujano selecciona el nivel en el que desea entrenarse: básico, medio, avanzado.</p>	<p>2.1. El sistema asigna la técnica de sutura a emplear según el nivel de entrenamiento elegido por el Médico Cirujano.</p>
<p>Sección “Iniciar Grapado”</p>	
<p>Flujo Normal</p>	
<p>Acción del Actor</p>	<p>Respuesta del Sistema</p>
<p>1. El Médico Cirujano selecciona el ejercicio de grapado.</p>	<p>1.1. El sistema carga el modelo de la escena de grapado</p> <p>1.2. El sistema crea y posiciona la cámara.</p> <p>1.3. El sistema crea y posiciona las luces.</p> <p>1.4. El sistema crea y posiciona las pinzas.</p> <p>1.5. El sistema crea las grapas.</p>
<p>2. El Médico Cirujano selecciona el nivel en el que desea entrenarse: básico, medio, avanzado.</p>	<p>2.1. El sistema asigna la cantidad de grapas a colocar según el nivel de entrenamiento elegido por el Médico Cirujano.</p>
<p>Poscondiciones</p>	<p>El ejercicio seleccionado por el Médico Cirujano se ha cargado correctamente.</p>

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

Tabla 2.5: Descripción textual del caso de uso Resetear Ejercicio.

Caso de Uso:	Resetear Ejercicio.	
Actores:	Médico Cirujano (inicia).	
Resumen:	El caso de uso inicia cuando el Médico Cirujano decide seleccionar un nivel de entrenamiento: Básico, Medio o Avanzado.	
Precondiciones:	El Médico Cirujano previamente autenticado. Debe haber iniciado un ejercicio.	
Referencias	R-5.1, R-5.2, R-5.3.	
Prioridad	Secundario.	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
1. El Médico Cirujano decide resetear el ejercicio en el que se está entrenando.	<p>1.1. El sistema actualiza la cantidad de veces que se ha reseteado el ejercicio.</p> <p>1.2. El sistema posiciona los elementos de la escena en las posiciones iniciales.</p>	
Flujo Alternativo		
	1.1. Si la cantidad de intentos es igual a 3 se cierra el ejercicio.	
Poscondiciones	Se han ubicado en sus posiciones iniciales lo elementos de la escena.	

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

Tabla 2.6: Descripción textual del caso de uso Emitir Evaluación.

Caso de Uso:	Emitir Evaluación.
Actores:	Usuario del Sistema (Inicia).
Resumen:	El caso de uso se inicia cuando el Médico Cirujano concluye el ejercicio y decide guardar la evaluación obtenida.
Precondiciones:	El Médico Cirujano debe haber terminado el ejercicio previamente seleccionado.
Referencias	R-29.1, R-29.2.
Prioridad	Secundario
Flujo Normal de Eventos	
Acción del Actor	Respuesta del Sistema
1. El Médico Cirujano decide guardar la evaluación obtenida en el ejercicio.	1.1. El sistema emite la evaluación teniendo en cuenta los errores cometidos en el ejercicio y la almacena en la Base de Datos.
2. El Médico Cirujano observa la evaluación y sale de ejercicio.	2.2. El sistema cierra el ejercicio y brinda la posibilidad al Médico Cirujano de seleccionar otro ejercicio en el que desee entrenarse.
Poscondiciones	El Médico Cirujano concluyó su entrenamiento y conoce los resultados alcanzados.

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

Tabla 2.7: Descripción textual del caso de uso Gestionar Evaluación.

Caso de Uso:	Gestionar Evaluación.	
Actores:	Profesor (Inicia).	
Resumen:	El caso de uso se inicia cuando el Profesor decide revisar la evaluación de un Médico Cirujano.	
Precondiciones:	El Profesor debe haber sido registrado previamente.	
Referencias	R-30.	
Prioridad	Secundario	
Flujo Normal de Eventos		
	Acción del Actor	Respuesta del Sistema
	1. El Profesor selecciona la opción “Revisar Evaluaciones” del Menú Principal.	1.1. El sistema muestra la interfaz que permite seleccionar el nombre del Médico Cirujano y el ejercicio del que se desea revisar la evaluación obtenida por el mismo.
	2. El Profesor selecciona el nombre del Médico Cirujano que desea conocer su evaluación.	2.1. El sistema adiciona al ComboBox los ejercicios en lo que el médico seleccionado se ha entrenado.
	3. El Profesor selecciona el ejercicio del que desea ver la evaluación.	3.1. El sistema muestra en una tabla la evaluación del médico en el ejercicio seleccionado por niveles.
Poscondiciones	El Profesor obtuvo el resultado alcanzado por Medico Cirujano en algún ejercicio.	

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

A continuación se presentarán las especificaciones a partir de la técnica de Escenarios siguiendo la distribución de los requisitos funcionales definidos en el análisis de requisitos (Ver Tabla 2.8):

Tabla 2.8: Escenarios definidos para la primera y segunda etapa de desarrollo del SimQ.

Escenarios	Requisitos Funcionales	Etapas de Desarrollo
Interactuar con la cámara	R-6.1, R-6.2.	E1
Focalizar Objetos	R-7.1, R-7.2, R-7.3, R-7.4, R-7.5, R-7.6, R-8.1, R-8.2, R-8.3, R-9.1, R-9.2, R-9.3, R-9.4, R-9.5.	E1
Seleccionar Objeto	R-10.1, R-10.2, R-11.1, R-11.2, R-11.3, R-11.4, R-11.5, R-12, R-13.1, R-13.2, R-13.3, R-13.4, R-13.5, R-14.1, R-14.2, R-14.3.	E1
Interactuar con las pinzas	R-10.1, R-10.2, R-15.1.	E1
Agarrar y Transferir Objeto	R-11.1, R-11.2, R-11.3, R-11.4, R-11.5, R-12, R-15.1, R-15.2, R-15.3, R-15.4, R-15.5, R-15.6, R-15.7, R-15.8, R-15.9, R-16.1, R-16.2, R-16.3, R-16.4, R-16.5.	E1
Entrenar Corte	R-20.1, R-20.2, R-20.3, R-21.1, R-21.2, R-21.3, R-21.4, R-21.5, R-22.1, R-22.2, R-22.3, R-22.4.	E2
Entrenar Sutura	R-23.1, R-23.2, R-23.3, R-23.4, R-23.5, R-24.1, R-24.2, R-24.3, R-24.4, R-24.5, R-24.6, R-24.7, R-25.1, R-25.2, R-25.3, R-25.4.	E2
Entrenar de Grapado	R-26.1, R-26.2, R-27.1, R-27.2, R-27.3, R-27.4, R-27.5, R-27.6, R-28.1, R-28.2, R-28.3.	E2

Tabla 2.9: Escenario Interactuar con la Cámara.

Título: Interactuar con la Cámara.
Objetivo: Interactuar con la cámara, de manera que se pueda efectuar movimientos en todas las direcciones posibles.
Resumen: El escenario se inicia cuando el Médico Cirujano mueve por primera vez la cámara con el objetivo de focalizar un objeto hasta manifestar los movimientos en las posibles direcciones.
Precondiciones: El Médico cirujano debe haber seleccionado previamente el ejercicio de la cámara.
Actores: Médico Cirujano, Profesor.
Episodios: <ol style="list-style-type: none">1. Animar el objeto que el Médico Cirujano debe enfocar con una luz parpadeante.2. Mover la cámara según los movimientos indicados por el Médico Cirujano.3. Mover luz junto con la cámara.
Excepciones: <p>Episodio 1: <u>Si</u> el sistema detecta que la cámara está estática (sin movimiento) mientras el objeto aun no está focalizado <u>entonces</u> calcula el tiempo que se mantiene estática e incrementa las estadísticas de tiempo de indecisión.</p> <p>Episodio 1: <u>Si</u> el sistema detecta que la cámara esta colisionando con la pared de intestino <u>entonces</u> aplica una fuerza física a la cámara en su centro de masa para que reaccione al choque alejándose de la pared del intestino de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.</p>
Pulsar sobre la imagen para ver el video que representa los requisitos involucrados en este escenario.

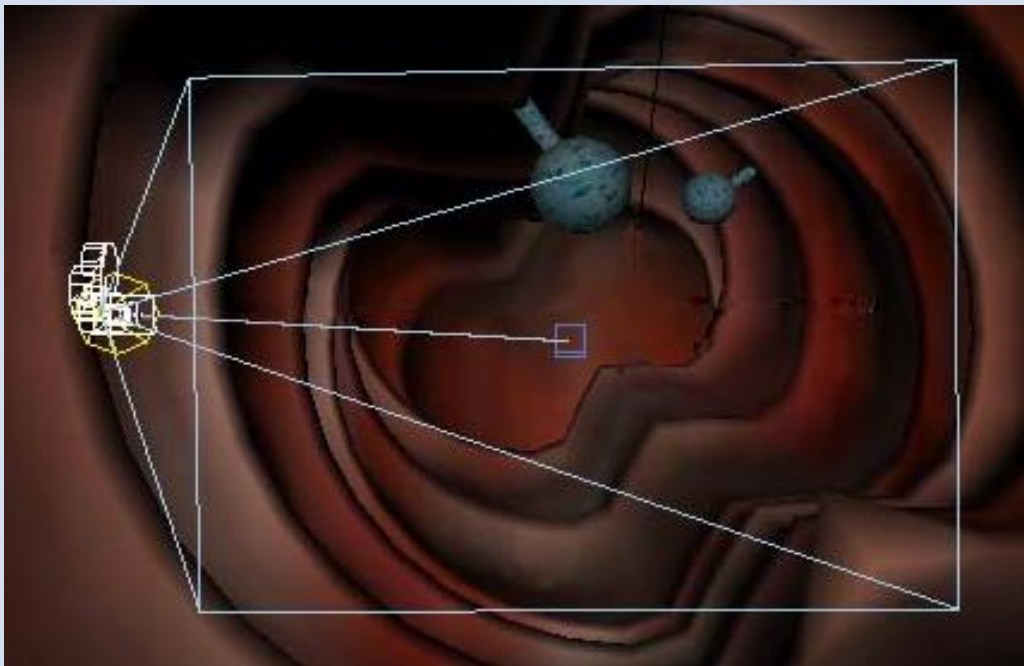


Tabla 2.10: Escenario Focalizar Objeto.

Título: Focalizar Objeto.

Objetivo: Focalizar el objeto que se encuentra animado con una luz parpadeante.

Resumen: El escenario se inicia cuando el Médico Cirujano focaliza un objeto.

Precondiciones: El Médico Cirujano debe haber seleccionado previamente el ejercicio de la cámara.

Actores: Médico Cirujano, Profesor.

Episodios:

1. Focalizar el objeto que está animado con una luz parpadeante.
2. Si el sistema detecta que el objeto está focalizado entonces anima la textura del objeto enfocado y activa el reloj de conteo regresivo.
3. Si el sistema detecta que la cuenta del reloj regresivo está en cero entonces devuelve al objeto su textura original, posteriormente elimina el objeto de la escena, decrementa la cantidad de objetos que aun no han sido focalizados e incrementar las estadísticas de

aciertos.

Excepciones:

Episodio 3: Si el sistema detecta que el objeto no está focalizado cuando el conteo del reloj regresivo aun no ha llegado a cero entonces incrementa las estadísticas de errores de focalización y desactiva el contador del reloj regresivo.

Episodio 3: Si el sistema detecta que la cámara está colisionando con el objeto entonces aplica una fuerza física a la cámara en su centro de masa para que reaccione al choque alejándose del objeto de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Pulsar sobre la imagen para ver el video que representa los requisitos involucrados en este escenario.

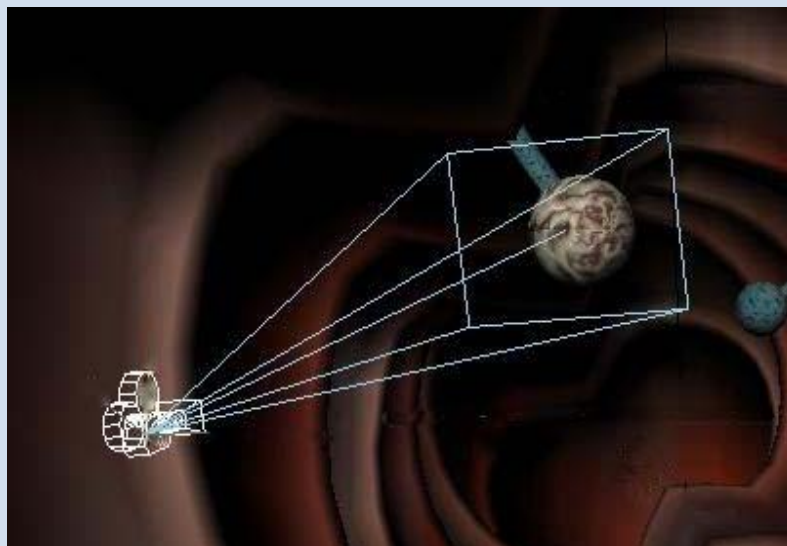


Tabla 2.11: Escenario Seleccionar Objeto.

Título: Seleccionar Objeto.

Objetivo: Seleccionar un objeto con la pinza correcta.

Resumen: El escenario se inicia cuando el Médico Cirujano activa una pinza con el objetivo de seleccionar un objeto hasta lograr su selección.

Precondiciones: El Médico Cirujano debe haber seleccionado previamente el ejercicio de selección.

Actores: Médico Cirujano, Profesor.

Episodios:

1. Seleccionar la pinza que se desea activar.
2. Mover la pinza activada según los movimientos indicados por el Médico Cirujano.
3. Si la pinza activada está colisionando con algún objeto y este se encuentre dentro del grupo de objetos que la pinza puede seleccionar entonces simular la desaparición del objeto de la escena, incrementa las estadísticas de aciertos y decrementa la cantidad de objetos que aún no han sido seleccionados.

Excepciones:

Episodio 2: Si el sistema detecta que la pinza activada está estática (sin movimiento) entonces incrementa las estadísticas de tiempo de indecisión.

Episodio 2: Si las pinzas están colisionando entre sí entonces aplica una fuerza física a la pinza que no se encuentra activada en su centro de masa para que reaccione al choque alejándose de la pinza activada de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Episodio 2: Si la pinza activada está colisionando con el torso entonces aplica una fuerza física a la pinza en su centro de masa para que reaccione al choque alejándose del torso de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Episodio 3: Si el objeto que esta colisionando con la pinza activada no se encuentra dentro del grupo de objetos que esta pinza puede seleccionar entonces simular explosión del objeto de forma que se elimine de la escena e incrementa estadísticas de errores por concepto de selección incorrecta.

Pulsar sobre la imagen para ver el video que representa los requisitos involucrados en este escenario.

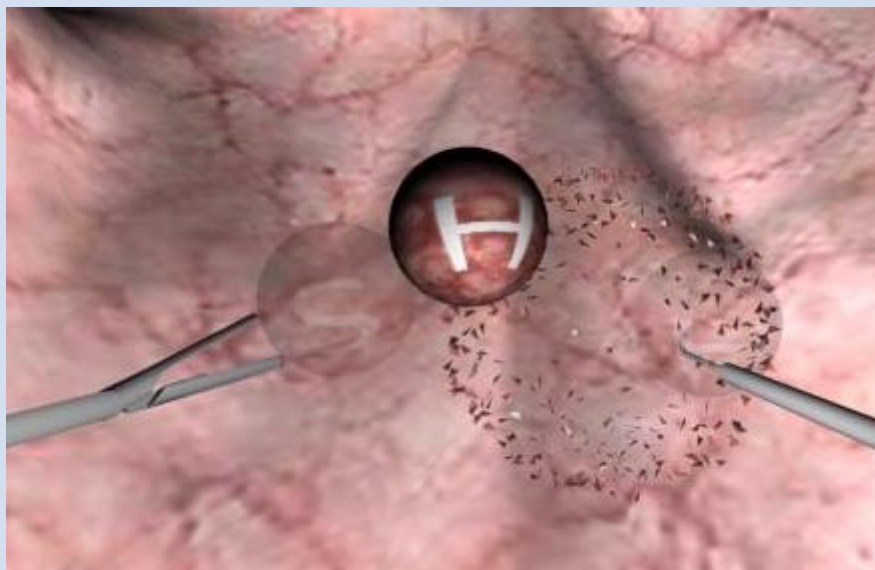


Tabla 2.12: Escenario Interactuar con las Pinzas.

Título: Interactuar con las Pinzas.

Objetivo: Interactuar con las pinzas efectuando movimientos en todas las direcciones posibles.

Resumen: El escenario se inicia cuando el Médico Cirujano activa una pinza con el objetivo de realizar algunos movimientos con ella incluyendo el movimiento de sus muelas.

Precondiciones: El Médico Cirujano debe haber seleccionado previamente el ejercicio de coordinación.

Actores: Médico Cirujano, Profesor.

Episodios:

1. Seleccionar la pinza que se desea activar.
2. Mover la pinza activada según los movimientos indicados por el Médico Cirujano.
3. Oprimir las teclas A o C en correspondencia con las acciones Abrir o Cerrar las muelas de las pinzas.

Excepciones:

Episodio 2: Si el sistema detecta que la pinza activada está estática (sin movimiento) entonces calcula el tiempo que la pinza activada se mantiene estática e incrementa las estadísticas de tiempo de indecisión.

Episodio 2: Si las pinzas están colisionando entre sí entonces aplica una fuerza física a la pinza que no se encuentra activada en su centro de masa para que reaccione al choque alejándose de la pinza activada de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Episodio 2: Si la pinza activada están colisionando con la escena entonces aplica una fuerza física a la pinza en su centro de masa para que reaccione al choque alejándose del torso de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Pulsar sobre la imagen para ver el video que representa los requisitos involucrados en este escenario.

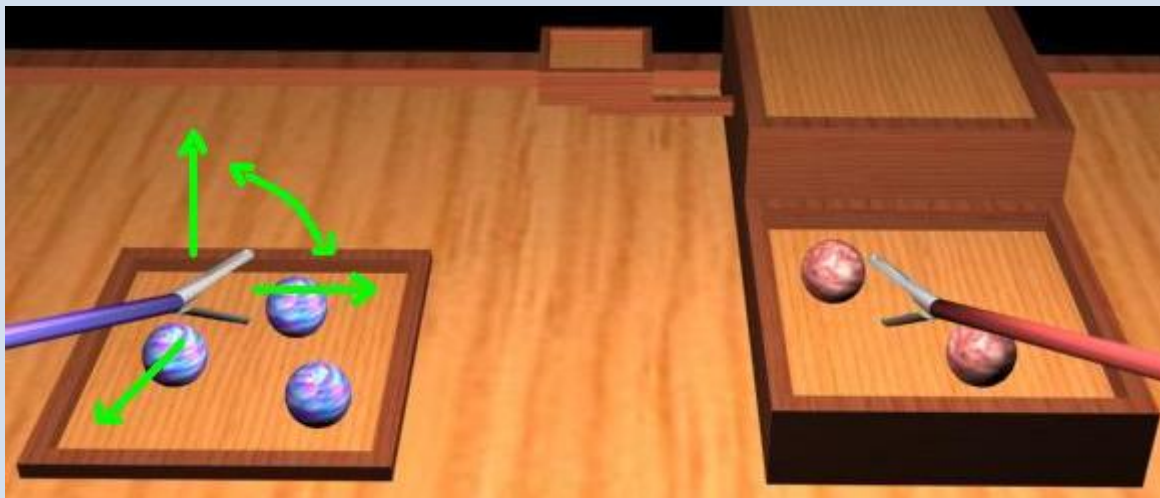


Tabla 2.13: Escenario Agarrar y Transferir Objeto.

Título: Agarrar y Transferir Objeto.

Objetivo: Agarrar un objeto y depositarlo en un recipiente.

Resumen: El escenario se inicia cuando una de la pinza agarra un objeto y lo transfiere hasta

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

depositarlo en la caja destino.

Precondiciones: El Médico Cirujano debe haber seleccionado previamente el ejercicio de coordinación.

Actores: Médico Cirujano, Profesor.

Episodios:

1. Si la pinza activada está colisionando con algún objeto, este se encuentre dentro del grupo de objetos que la pinza puede seleccionar y el ángulo de amplitud entre las muelas de las pinzas es mayor de 45° entonces la pinza activada selecciona el objeto en cuestión.
2. Mover las pinzas con el objeto seleccionado en dirección del área donde se desea depositar.
3. Abrir las muelas de las pinzas para depositar el objeto en el área de destino.
4. Si el objeto es depositado correctamente entonces el sistema decrementa la cantidad de objetos que aun no han sido transferidos e incrementa las estadísticas de aciertos.

Excepciones:

Episodio 1: Si el sistema detecta que la pinza activada está colisionando con algún objeto, este se encuentre dentro del grupo de objetos que la pinza puede seleccionar y el ángulo de amplitud entre las muelas de las pinzas es menor de 45° entonces aplica una fuerza física al objeto en su centro de masa para que reaccione al choque alejándose de la pinza de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Episodio 1: Si la pinza activada está colisionando con algún objeto y este no se encuentre dentro del grupo de objetos que la pinza puede seleccionar entonces incrementar estadísticas de errores de por concepto de selección incorrecta.

Episodio 4: Si al depositar el objeto cae fuera en el área de destino entonces incrementa la cantidad de objetos caídos.

Pulsar sobre la imagen para ver el video que representa los requisitos involucrados en este escenario.

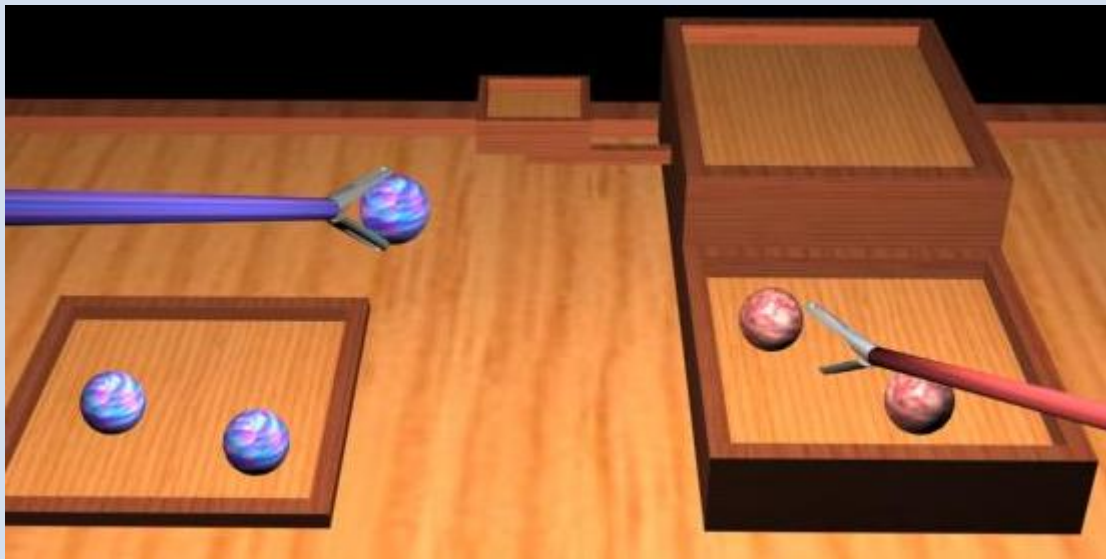


Tabla 3: Escenario Entrenar Corte.

Título: Entrenar Corte.

Objetivo: Realizar un corte.

Resumen: El escenario se inicia cuando el Médico Cirujano mueve la tijera con el objetivo de realizar un corte.

Precondiciones: El Médico Cirujano debe haber seleccionado previamente el ejercicio de corte.

Actores: Médico Cirujano, Profesor.

Episodios:

1. Activar la tijera.
2. Mover la tijera según los movimientos indicados por el Médico Cirujano.
3. Realizar corte según la trayectoria definida.
4. Si el corte realizado se ajusta a la trayectoria definida entonces incrementar estadísticas de aciertos.

Excepciones:

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

Episodio 2: Si el sistema detecta que la herramienta activada se encuentran estáticas (sin movimiento) entonces incrementa las estadísticas de errores por concepto de tiempo de indecisión.

Episodio 2: Si el sistema detecta que la tijera y la pinza están colisionando entre sí entonces aplica una fuerza física a la pinza en su centro de masa para que reaccione al choque alejándose de la tijera de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Episodio 4: Si el corte realizado no se ajusta a la trayectoria definida entonces incrementar estadísticas de errores por concepto de corte incorrecto.

Pulsar sobre la imagen para ver el video que representa los requisitos involucrados en este escenario.



Tabla 2.15: Escenario Entrenar Sutura.

Título: Entrenar Sutura.

Objetivo: Realizar sutura.

Resumen: El escenario se inicia cuando el Médico Cirujano mueve la aguja con el objetivo de realizar una sutura.

Precondiciones: El Médico Cirujano debe haber seleccionado previamente el ejercicio de sutura.

Actores: Médico Cirujano, Profesor.

Episodios:

1. Activar el porta-agujas.
2. Mover el porta-agujas según los movimientos indicados por el Médico Cirujano.
3. Introducir y sacar la aguja a través de los puntos que aparecen marcados en el intestino.
4. Si la aguja siguió la trayectoria de los puntos marcados en el intestino entonces incrementar las estadísticas de aciertos.
5. Realizar tipo de nudo correspondiente con la técnica de sutura asignada al nivel de entrenamiento en cuestión.
6. Si el nudo realizado está en correspondencia con la técnica de sutura asignada al nivel de entrenamiento en cuestión entonces incrementar estadísticas de aciertos.
7. Cortar el hilo con la tijera.

Excepciones:

Episodio 2: Si el sistema detecta que la herramienta activada se mantiene estática (sin movimiento) entonces incrementa las estadísticas de errores por concepto de tiempo de indecisión.

Episodio 2: Si el sistema detecta que el porta-agujas y la pinza están colisionando entre sí entonces aplica una fuerza física a herramienta activada en su centro de masa para que reaccione al choque alejándose de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Episodio 2: Si el sistema detecta que la tijera y la pinza están colisionando entre sí entonces aplica una fuerza física a herramienta activada en su centro de masa para que reaccione al choque alejándose de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Episodio 6: Si el nudo realizado no está en correspondencia con la técnica de sutura asignada al nivel de entrenamiento en cuestión o la aguja siguió la trayectoria de los puntos marcados entonces incrementar estadísticas de errores cometidos en el ejercicio.

Pulsar sobre la imagen para ver el video que representa los requisitos involucrados en este escenario.

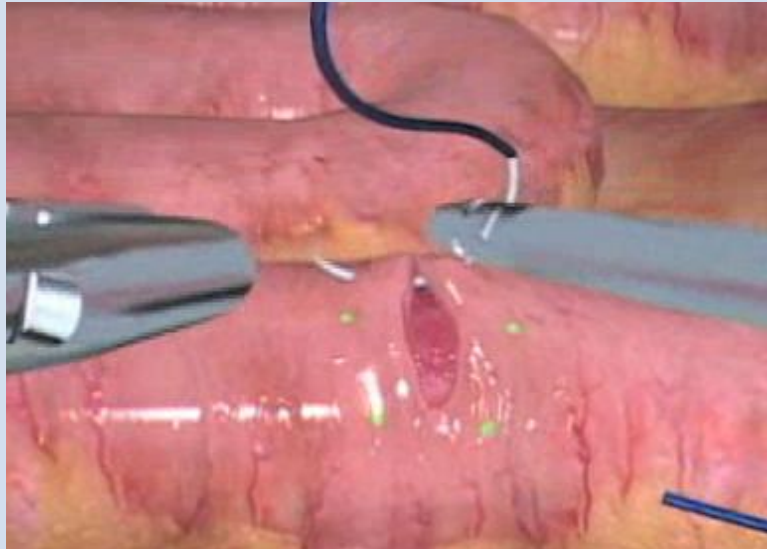


Tabla 2.16: Escenario Entrenar Grapado.

Título: Entrenar Grapado.

Objetivo: Realizar un grapado.

Resumen: El escenario se inicia cuando el Médico Cirujano mueve la pinza con el objetivo de colocar una grapa.

Precondiciones: El Médico Cirujano debe haber seleccionado previamente el ejercicio de grapado.

Actores: Médico Cirujano, Profesor.

Episodios:

1. Mover la grapadora según los movimientos indicados por el Médico Cirujano.
2. Colocar grapa.
3. Si la grapa se colocó en el sitio correcto entonces incrementar estadísticas de aciertos.

Excepciones:

CAPÍTULO II: ELICITACIÓN, ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS

Episodio 1: Si el sistema detecta que la grapadora está estática (sin movimiento) entonces incrementa las estadísticas de errores por concepto de tiempo de indecisión.

Episodio 1: Si el sistema detecta que la grapadora y la pinza están colisionando entre sí entonces aplica una fuerza física a la pinza en su centro de masa para que reaccione al choque alejándose de forma que desaparezca la colisión e incrementa las estadísticas de errores por concepto de colisión.

Episodio 3: Si la grapa no se colocó en el sitio correcto entonces incrementar cantidad de errores por concepto de grapado incorrecto.

Pulsar sobre la imagen para ver el video que representa los requisitos involucrados en este escenario.



CAPÍTULO 3: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA.

La primera parte de este capítulo estará dedicada a la propuesta de la actividad validación de requisitos, la cual permite probar los requisitos antes de que comience a diseñarse e implementarse el sistema, detectar errores y deficiencias. Con este propósito se desarrollará en el presente acápite un prototipo no funcional del SimQ cuyo objetivo será comprobar los requisitos definidos previamente con el cliente. En la segunda parte del capítulo se realizará una valoración de la propuesta desarrollada a partir del criterio del equipo de desarrolladores del proyecto SimQ, así como stakeholders.

3.1. Propuesta de Validación de Requisitos para el SimQ.

Cuando la especificación de requisitos de un software es terminada es de vital importancia garantizar que los requisitos definidos sean los correctos, así como el hecho de que estas especificaciones sean completas. Es por ello que luego de realizar las actividades de elicitación, análisis y especificación se debe verificar que los requisitos que aparecen reflejados en el documento satisfagan las necesidades del cliente, con el fin de evitar que se detecten errores en etapas posteriores del desarrollo del software derivadas de una incorrecta especificación de los requisitos. En función de lograr este objetivo se propone el uso de la técnica de validación: prototipado y se procederá a construir un prototipo no funcional del SimQ que permitirá comprobar los requisitos definidos previamente con el cliente. Este prototipo no funcional proporcionará al usuario una visión del producto final, a la vez que clientes y desarrolladores pueden identificar funcionalidades incorrectas.

3.1.1. Prototipo No Funcional.

El prototipo no funcional constará de varias interfaces interconectadas entre sí, de manera que cuando se pulse algún botón de la interfaz la aplicación responderá a dicha acción mostrando la interfaz correspondiente.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

Al iniciarse la aplicación se cargará en pantalla una interfaz que le permitirá al usuario autenticarse y de esta forma se le concedan los permisos correspondientes según su rol (Ver Figura 3.1)



Figura 3.1: Interfaz autenticar usuario.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

Si el usuario autenticado tiene asignado el rol Médico Cirujano, se presentará una interfaz en la cual el médico, al desplegar el menú, podrá seleccionar el ejercicio en el que se desea entrenar (Ver Figura 3.2).

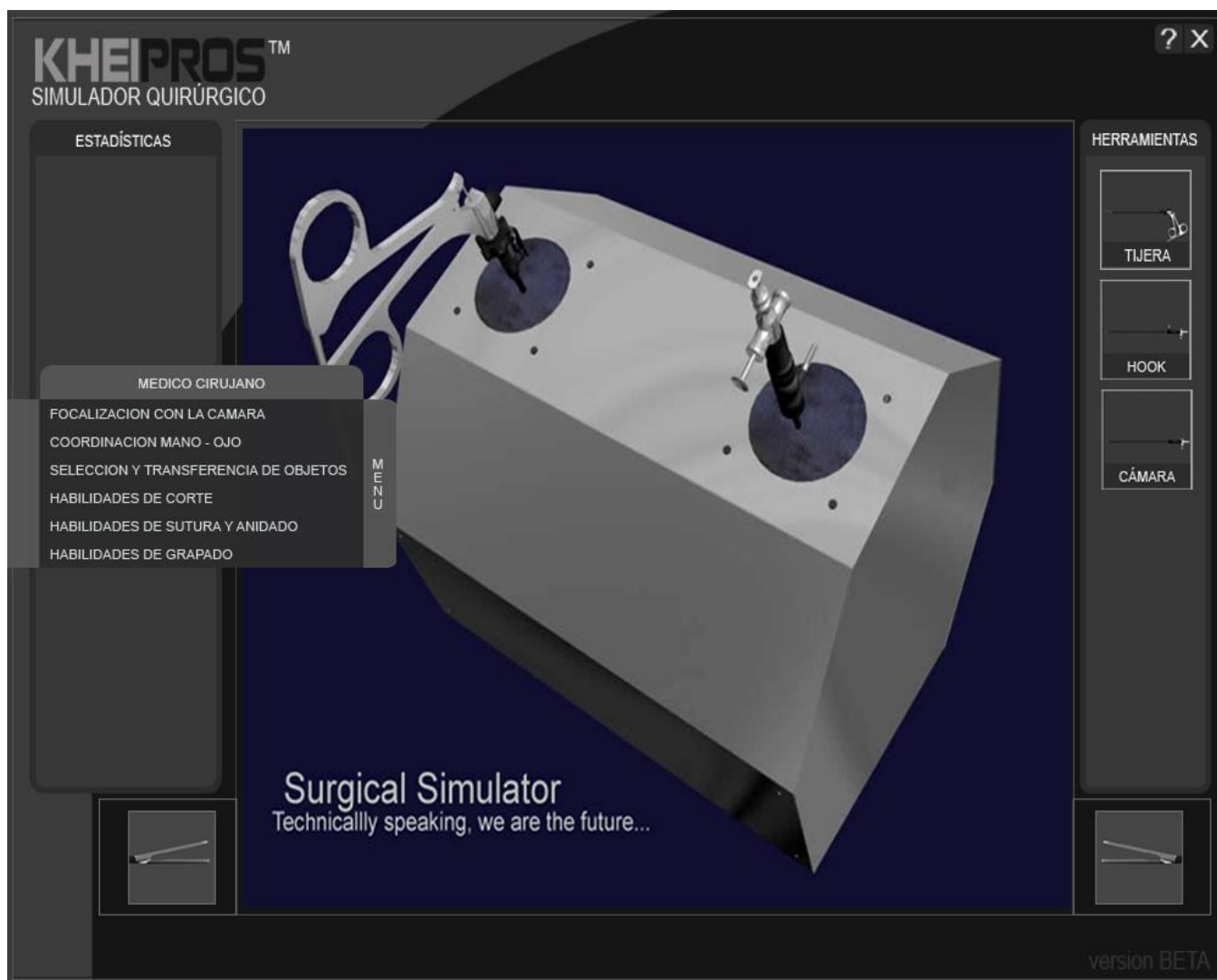


Figura 3.2: Interfaz Principal del Médico Cirujano.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

Cuando el médico seleccione un ejercicio se cargará la escena correspondiente con un menú que permitirá seleccionar el nivel de dificultad en el que desea entrenarse (Ver Figura 3.3).



Figura 3.3: Interfaz seleccionar nivel de entrenamiento.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

Si el usuario que se autentica tiene asignado el rol Profesor se mostrará una interfaz con la opción de entrenarse en algunos de los ejercicios, al igual que un Médico Cirujano y además la opción revisar las evaluaciones obtenidas por los médicos que ya se han entrenado en dichos ejercicios (Ver Figure 3.4).

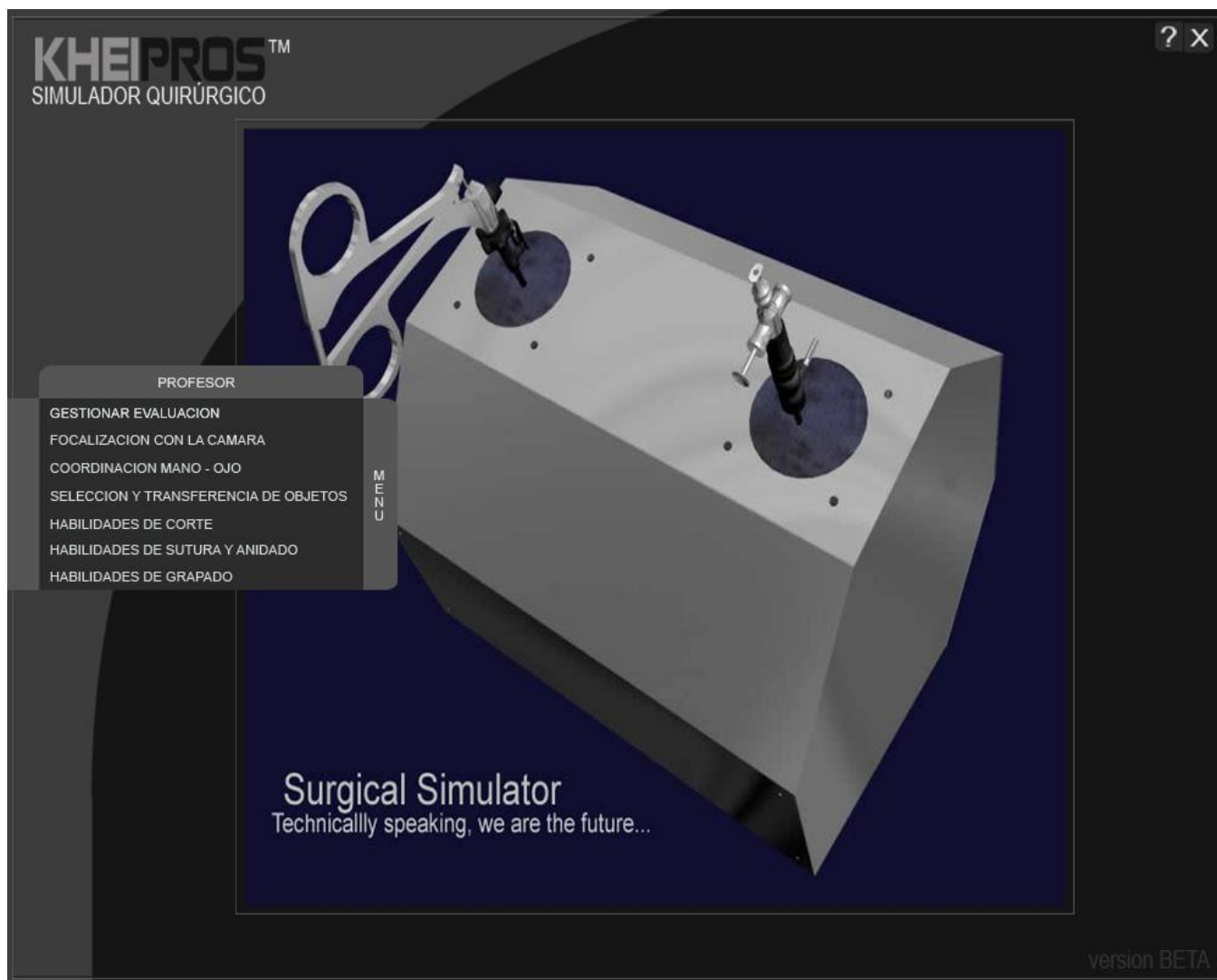


Figure 3.4: Interfaz principal del profesor.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

Cuando el profesor decide revisar la evaluación de algún médico se carga en pantalla una interfaz que permitirá seleccionar el nombre del médico que se desea consultar sus evaluaciones, así como el ejercicio que se revisará (Ver Figura 3.5).

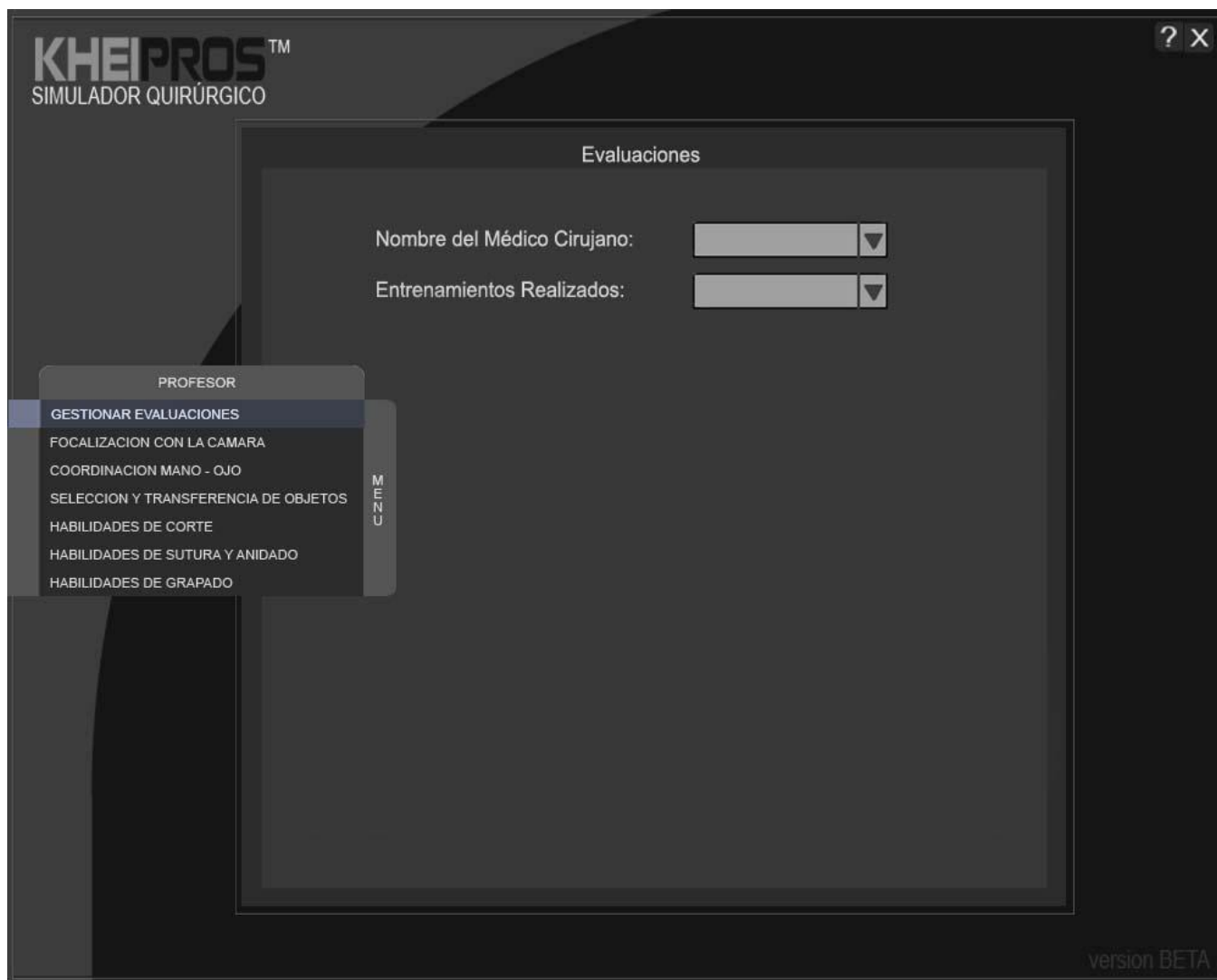


Figura 3.5: Interfaz gestionar evaluaciones.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

Si el usuario que se autentica tiene asignado el rol Administrador, la interfaz que se presentará, brindará la posibilidad de escoger mediante el menú si se desea Registrar, Modificar o Eliminar los datos de un usuario (Ver Figura 3.6).



Figura 3.6: Interfaz principal de la sesión administrador.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

El administrador puede seleccionar la opción registrar un nuevo usuario del menú, y se mostrará una interfaz donde se brinda la posibilidad de insertar los datos, así como el rol que dicho usuario jugará en el sistema (Ver Figura 3.7).

The screenshot shows the 'KHEIPROS™ SIMULADOR QUIRÚRGICO' application window. On the left, a 'MENU' sidebar contains the following options: ADMINISTRADOR, REGISTRAR USUARIO (highlighted), MODIFICAR USUARIO, and ELIMINAR USUARIO. The main area is titled 'Datos del Usuario' and contains the following fields:

- Nombre:
- Apellidos:
- Usuario:
- Contraseña:
- Dirección Particular:
- Teléfono:
- Centro de Trabajo:
- Años de Experiencia:
- Asignar Rol:
 - Médico Cirujano
 - Profesor
 - Administrador

At the bottom right, there is a 'REGISTRAR' button and the text 'version BETA'.

Figura 3.7: Interfaz registrar usuario.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

Si el administrador selecciona la opción modificar usuario del menú, se presentará una interfaz en la cual se cargará una tabla con el nombre de todos los usuarios del sistema permitiendo la modificación de los datos pertenecientes al usuario seleccionado (Ver Figura 3.8).



Figura 3.8: Interfaz modificar datos del usuario.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

Una vez que el administrador seleccione la opción eliminar usuario del menú, se presentará una interfaz en la cual se cargará una tabla con el nombre de todos los usuarios del sistema (Ver Figura 3.9).

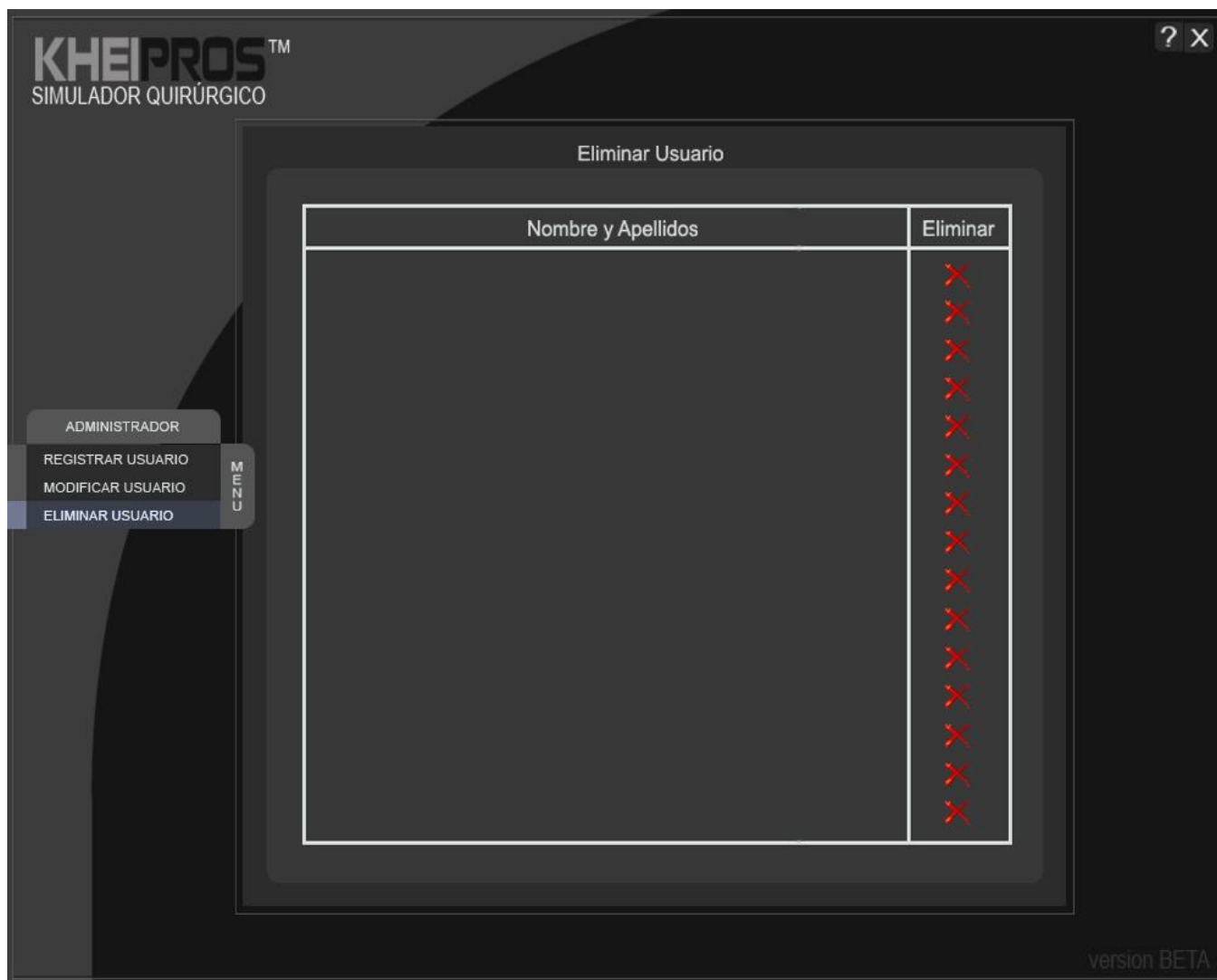


Figura 3.9: Eliminar datos del usuario.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

Para realizar la validación de los requisitos a partir del prototipo construido se propone establecer una comparación de cada uno de los requisitos definidos en la lista de requisitos realizada en la etapa de análisis con cada una de las funcionalidades que se representan en el prototipo, de manera que el usuario puedan validar, chequear que las funcionalidades representadas son realmente las esperadas, las que se corresponden con sus necesidades. Luego de establecer dicha comparación se emitirá un documento de no conformidades donde se resuman las deficiencias detectadas.

3.2. Valoración de la Propuesta de Solución Desarrollada.

Concluidas las cuatro etapas de desarrollo de requisitos definidas en la propuesta aplicada al SimQ, se procede a realizar una valoración que permita valorar la importancia de la propuesta teniendo en cuenta dos aristas: la opinión de los desarrolladores del proyecto y la del cliente. Teniendo en cuenta que el proyecto SimQ actualmente no cuenta con un cliente concretamente definido la valoración de la propuesta se realizará a partir del criterio de algunos stakeholders que han estado involucrados en la realización del proyecto desde sus inicios y que en algún momento pudieron tener contactos mínimos con un cliente inicial. Para materializar la valoración correspondiente a la primera arista se encuestarán un total de diez desarrolladores del proyecto teniendo en cuenta el desarrollo de la primera etapa del SimQ a partir del uso de la propuesta de desarrollo de requisitos utilizada. En el caso de la segunda arista se encuestarán cuatro stakeholders que valorarán la propuesta con respecto a su entendimiento y correspondencia de las funcionalidades esperadas tanto para la primera como para la segunda etapa.

3.2.1. Valoración por parte de los Desarrolladores.

Para valorar los resultados a partir del criterio de los desarrolladores del proyecto se tuvieron en cuenta algunos aspectos que reflejan la eficiencia de la propuesta realizada a partir de la solución a algunos de los problemas existentes antes de su aplicación. A continuación se enuncian dichos aspectos.

- A:** Especificación de requisitos incompleta.
- B:** Necesidad por parte de los desarrolladores de consultar otros documentos debido a una insuficiente la especificación de requisitos.
- C:** Influencia de la propuesta desarrollada en la disminución de los cambios en los requerimientos y sus especificaciones.
- D:** Medida en que se manifestó la falta de claridad en los objetivos.
- E:** Influencia de la propuesta desarrollada en el logro de una mejor comunicación con el equipo de desarrollo.

CAPÍTULO III: VALIDACIÓN DE REQUISITOS Y VALORACIÓN DE LA PROPUESTA

En la encuesta realizada a 10 desarrolladores del proyecto (Ver Anexo 5), el 80% de los encuestados manifiestan que la especificación de requisitos realizada es completa, lo que propició que sólo el 30% de éstos tuvieran en algún momento la necesidad de realizar búsquedas en internet o consultar otras bibliografías debido a una insuficiente especificación de requisitos (Ver Figura 3.10).

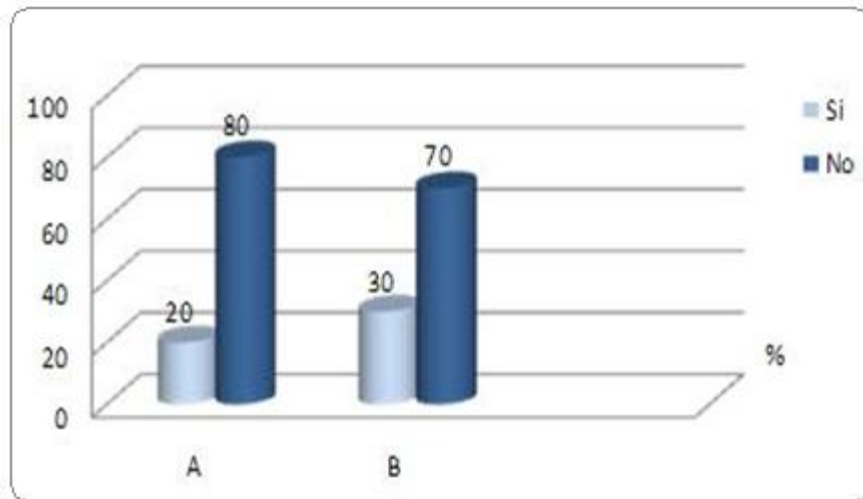


Figura 3.10: Factores que reflejan la completitud de la especificación de los requisitos.

Estos resultados muestran que con la aplicación de la propuesta se logró un mejor desarrollo de los requisitos propiciando un mayor entendimiento, por parte de los desarrolladores, de las funcionalidades a implementar. Sin embargo también se evidencia que aún existen desarrolladores, aunque son la minoría, que tuvieron que auxiliarse de fuentes externas para implementar las funcionalidades, por lo que se debe ganar más claridad en la especificación de los requisitos de las próximas etapas de desarrollo del simulador.

Por otra parte se obtuvo que el 80% de los encuestados consideran, en alta medida, que la aplicación de la propuesta desarrollada tuvo una influencia positiva en la disminución de las solicitudes de cambio en los requerimientos y sus especificaciones, además de facilitar la implementación de las funcionalidades del SimQ y solo el 20 % lo consideran en un nivel medio (Ver Figura 3.11).



Figura 3.11: Disminución de los cambios en los requerimientos y sus especificaciones.

De igual forma el 90% de ellos opina, en alta medida, que se logró una mejor comunicación entre el equipo de desarrollo y los analistas y solo el 10% lo plantean en un nivel medio (Ver Figura 3.12).



Figura 3.12: Comunicación con el equipo de desarrollo.

En la siguiente gráfica se muestra que el 80% de los encuestados manifiestan, en un nivel alto, que la aplicación de la propuesta le ha proporcionado una fácil implementación de las funcionalidades del proyecto lo que evidencia la claridad en los objetivos. Solo el 20% lo plantean en un nivel medio. (Ver Figura 3.13)

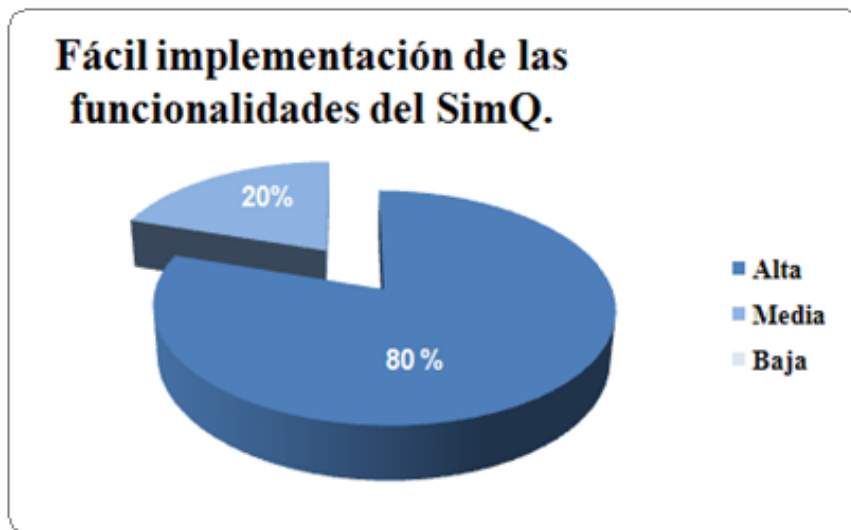


Figura 3.13: Facilidad en la implementación de las funcionalidades del Simulador a partir de la propuesta.

3.2.2. Valoración por parte de los Stakeholders.

Para valorar los resultados a partir del criterio de los stakeholders en el proyecto se tuvieron en cuenta algunos aspectos que reflejan la eficiencia de la propuesta realizada a partir de su entendimiento y correspondencia de las funcionalidades. A continuación se enuncian dichos aspectos.

- A:** Existencia de buena comunicación con los ingenieros de software.
- B:** Influencia de la propuesta desarrollada en el logro de una adecuada comunicación.
- C:** Facilidad de entendimiento y alta precisión en las características y funcionalidades definidas para el desarrollo del producto solicitado.
- D:** Disminución de las solicitudes de cambio en los requerimientos y sus especificaciones.

En la encuesta realizada a cuatros stakeholders (*Ver Anexo 6*) se evidenció que el 100% de los encuestados afirman haber logrado una adecuada comunicación con los ingenieros de requisitos y una disminución de las solicitudes de cambio en los requisitos y sus especificaciones (*Ver Figura 3.14*).

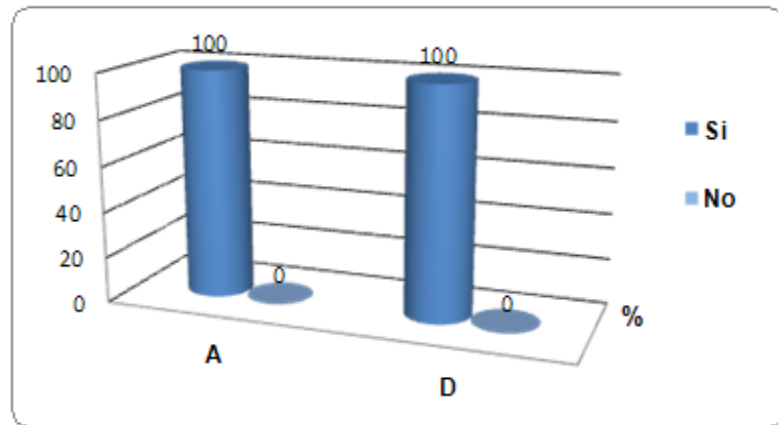


Figura 3.14: Adecuada comunicación y disminución de las solicitudes de cambio en los requerimientos y sus especificaciones.

El 90% de los encuestados consideran, en alta medida, que la aplicación de la propuesta desarrollada tuvo una influencia positiva en el logro de una adecuada comunicación (Ver Figura 3.15).

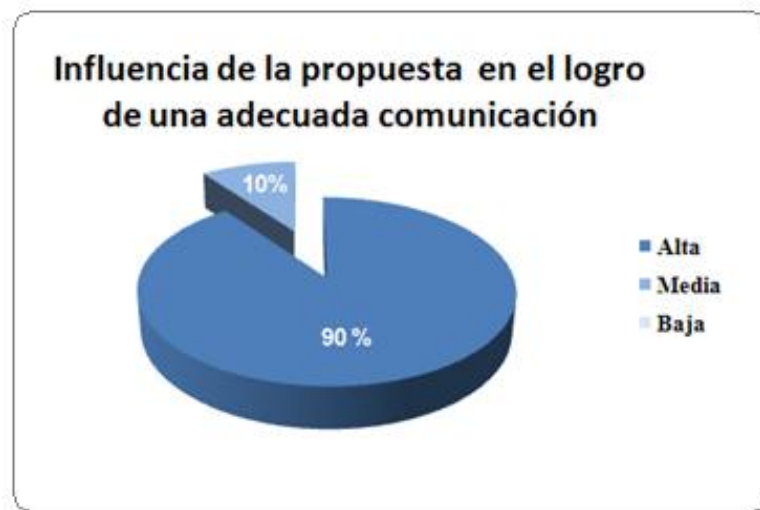


Figura 3.15: Influencia de la propuesta en el logro de una adecuada comunicación.

Se obtuvo además, que el 90% de los encuestados afirman, en alta medida, que la propuesta proporciona un buen entendimiento y alta precisión en las características y funcionalidades definidas para el desarrollo del Simulador y sólo un 10% lo consideran en un nivel medio (Ver Figura 3.16).

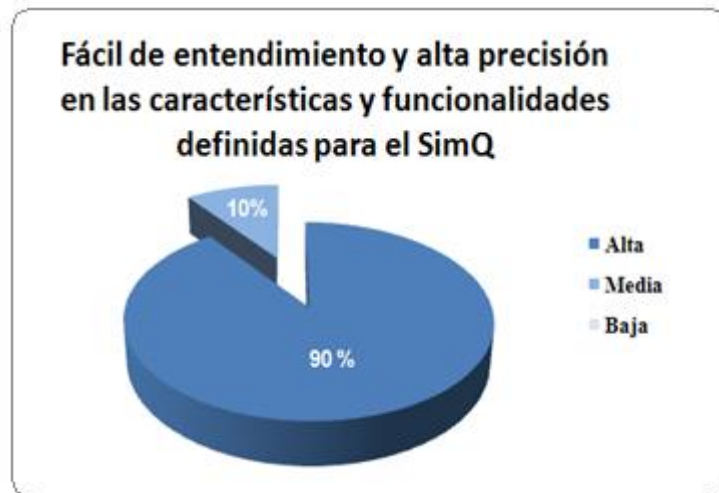


Figura 3.16: Entendimiento y precisión en las características y funcionalidades definidas.

Se puede concluir afirmando que la propuesta de solución desarrollada tiene, según la opinión de los desarrolladores y Stakeholder, un nivel de aceptación satisfactorio ya que se logró solucionar, en gran medida, los problemas existentes inicialmente.

CONCLUSIONES.

Con la culminación de este trabajo se obtuvo una propuesta de desarrollo de requisitos a aplicar en el SimQ lográndose de esta forma dar cumplimiento al objetivo planteado ya que:

1. Se realizó un estudio del estado del arte acerca del desarrollo de los Simuladores Quirúrgicos y de la Ingeniería de Requisitos.
2. Se investigó además el proceso de desarrollo de requisitos en simuladores cubanos señalándose los principales problemas detectados en esta área.
3. Se definieron las técnicas adecuadas a utilizar en cada una de las etapas que comprenden el desarrollo de requisitos del proyecto Simulador Quirúrgico y a su vez se aplicó la propuesta en las dos primeras etapas de desarrollo de dicho proyecto.
4. Se obtuvo un prototipo no funcional del Simulador Quirúrgico como resultado del empleo de la técnica de prototipado, que permitió la validación de los requisitos definidos en las etapas iniciales.
5. Se realizó una valoración de la propuesta a partir del criterio de desarrolladores del proyecto y stakeholder obteniéndose un alto grado de aceptación de la propuesta.

RECOMENDACIONES.

A partir de los resultados obtenidos en este trabajo se recomienda:

1. Realizar la validación de requisitos a partir de la construcción de una lista de chequeo que permita validar cada uno de los requerimientos definidos en la etapa de análisis con las funcionalidades representadas en el prototipo no funcional creado, generando un documento de no conformidades como se ha establecido en este trabajo.
2. Darle continuidad a la propuesta desarrollada lográndose una especificación más detallada que proporcione mayor claridad en la descripción de los requisitos para las próximas etapas de desarrollo del Simulador Quirúrgico.
3. Hacer extensible la propuesta desarrollada a proyectos de Realidad Virtual que posean características similares a las del Simulador Quirúrgico.

REFERENCIAS BIBLIOGRÁFICAS.

1. **Ávila, Lourdes García, Sánchez, Leidy Fernández y Robles, Esperanza Aguillón. 2007.** Procedimiento para el desarrollo del proceso de ingeniería de requisitos en un proyecto software (PROCIR). [En línea] 2007. http://www.informaticahabana.com/evento_virtual/files/CAL042.
2. **Bahamonde, J y Rossel, R. 2003.** *Un acercamiento a la Ingeniería de Requisitos. Universidad técnica Federico Santa María.* 2003.
3. **Borland. 2006.** *Effective Requirements Definition and Management, Improves System Communication.* 2006.
4. **Brooks, Frederick P,1987.** *Essence and Accident in Software Engineering.* 1987.
5. **Canós, José H, Letelier, Patricio y Penadés, M^a Carmen. 2000.** *Metodologías Ágiles en el Desarrollo de Software.* 2000.
6. **Chaves, Michael Arias. 2006.** La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de Software. 7 de julio de 2006. 2006.
7. **Corporation, S. 2004.** Simulab. [En línea] 2004. <http://www.simulab.com/AboutUs.htm>.
8. **Duran, A. 2002.** *Metodología para la elicitación de requisitos de software. Sevilla, Universidad de Sevilla. : s.n.* 2002.
9. **Endosurgery, SECLA. 2004.** Telecirugía: La Revolución Silenciosa de los Robots. [En línea] Octubre-diciembre de 2004. <http://www.seclaendosurgery.com/seclan9/prart01.htm>.
10. **Escalona, M y Koch, N. 2002.** *Ingeniería de requisitos en aplicaciones para la web. Un estudio comparativo. s.l. : Universidad de Sevilla.* 2002.
11. **Facultad de Ciencias Exactas. 2007.** Ingeniería de requisitos. Escenarios: Notas. [En línea] 11 de Octubre de 2007. http://www.exa.unicen.edu.ar/catedras/ingrequi/index_archivos/Notas-Escenarios.pdf.
12. **Fernández, Gerardo Escribano. 2002.** *Introducción a Extreme Programming. Ingeniería del Software II.* 2002.

13. **García, E, y otros. 2005.** Sobre Realidad, Surrealismo y realidad virtual. IES Príncipe de Asturias de Lorca (Murcia). [En línea] 2005. www.lacavernadeplaton.com/articulosbis/sobrealidad0506.htm.
14. **Hadad, Graciela, Doorn, Jorge H y Kaplan, Gladys N. 1999.** Enfoque Middle-Out en la Construcción e Integración de Escenario. [En línea] 1999. http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER99/hadad.pdf.
15. **Ibarreche, D y Macías, M. 2007.** *Modulo de seguimiento de terrenos para sistemas de realidad virtual. Trabajo de Diploma. Universidad de Ciencias informáticas.* 2007.
16. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000.** *El Proceso Unificado de Desarrollo de Software. s.l. : Pearson Education, S. A.* 2000.
17. **Jhon, F y Von, D. 2006.** "Virtual Reality, VPL Research". [En línea] 2006. <http://www.icc.uji.es/asignatura..>
18. **Larman, C. 1999.** *UML y patrones. Introducción al análisis y diseño orientado a objetos. s.l. : Prentice Hall Hispanoamericana, S.A. .* 1999.
19. **Leffingwell, D y Widrin, D. 2003.** *Managing Software requirements: A Use case Approach, Second Edition. SI: Addison Wesley.* 2003.
20. **Monserrat, Carlos, López, Óscar y Alcañíz, Mariano. 2004.** Especial: Tecnologías de Simulación y Planificación Quirúrgica. Estado del Arte en Simulación Quirúrgica. [En línea] 2004.
21. **Pacheco, Y y Díaz, A. 2007.** *Propuesta de Algoritmo para la Simulación de las explosiones aéreas para los simuladores de tiro. Trabajo de Diploma. Universidad de Ciencias Informáticas.* 2007.
22. **Pérez, Karina. 2007.** *Modelo de referencia de IR para proyectos de Bioinformática. Tesis para optar por el grado científico de Master en Ciencias.* 2007.
23. **Pérez, Lourdes. 2007.** Conducir, tirar y volar en realidad virtual. Artículo del periódico Granma. [En línea] 18 de Junio de 2007. <http://granma.co.cu/2007/06/08/nacional/artic01.html>.

- 24. Presman, R. 2002.** *Ingeniería de Software. Un enfoque práctico . s.l. : Quinta Edición. España: Mcgraw-Hill, 2002.*
- 25. Pressman, R. S. 2005.** *Ingeniería de software: Un enfoque práctico. s.l. : 5ta. La Habana, Editorial Félix Varela, 2005.*
- 26. Pressman, Roger S. 2006.** *Ingeniería de Software. Un enfoque práctico. s.l. : Sexta Edición. México: McGraw-Hill. 2006.*
- 27. Rangel, A. 2007.** *Análisis comparativo de técnicas de obtención de requerimientos para el módulo de facturación del aplicativo gestasoft hospitalario para Imsalud. Trabajo de Diploma Universidad de Pampolona. Colombia : s.n. 2007.*
- 28. Rational Unified Process. 2003.** 2003.
- 29. Ridao, Marcela, Doorn, Jorge y Leite, Julio César Sampaio do Prado. 2000.** *Uso de Patrones en la Construcción de Escenarios. 2000.*
- 30. Robertson, S y Robertson, J. 2006.** *Matering the Requirements process Second Edition. SI: Addison Wesley Professional. 2006.*
- 31. Stern, J, Zeltser, IS y Pearle, MS. 2007.** *Simbionix. Compañía productora de simuladores médicos. Department of Urology. . [En línea] 2007. <http://www.simbionix.com..>*
- 32. Villanueva, Isabel, Sánchez, Juan y Pastor, Óscar. 2005.** *Elicitación de requisitos en sistemas de gestión orientados a procesos. [En línea] 21 de Junio de 2005 de Junio de 2005. http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER05/isabel_villanueva.pdf.*
- 33. Wiegers, K. 2003.** *Software Requirements. s.l. : Second Edition. 2003.*
- 34. Wiegers, Karl E. 2006.** *More About Software Requirements: Thorny Issues and Practical Advice. 2006.*

BIBLIOGRAFÍAS CONSULTADAS.

1. **Borland. 2006.** EFFECTIVE REQUIREMENTS DEFINITION AND MANAGEMENT IMPROVES SYSTEMS AND COMMUNICATION,. [En línea] Abril de 2006. www.borland.com/resources/en/pdf/solutions/rdm_whitepaper.pdf.
2. **—. 2006.** MITIGATING RISK WITH EFFECTIVE REQUIREMENTS ENGINEERING. [En línea] Abril de 2006. www.borland.com/resources/en/pdf/white_papers/mitigating_risk_with_effective_requirements_engineering.pdf .
3. **Gómez Sánchez, Pio Iván. 2003.** *Uso de simuladores y otras ayudas educativas en medicina.* 2003.
4. **Guckenheimer, Sam y Perez, Juan J. 2006.** *Software Engineering with Microsoft Visual Studio Team System.* s.l. : Addison Wesley, 2006.
5. **Ibarra Fernández, Antonio Jose, Fernández Beltrán, Francisco y Gil Hermoso, María de los Remedios. 2007.** Suturas. [En línea] 2007. <http://www.eccpn.aibarra.org/temario/seccion8/capitulo136/capitulo136.htm>.
6. **IEEE. 1998.** Guía para el desarrollo de Especificaciones de Requirimientos de Sistema. 1998.
7. **Jeffries, Ron, Anderson, Ann y Hendrickson, Chet. 2000.** *Extreme Programming Installed.* s.l. : Addison Wesley , 2000.
8. **Laguna, M^a P, Lagerveld, B y Rosette, J. de la. 2005.** TÁCTICAS Y TRUCOS ENDOUROLÓGICOS EN LAPAROSCOPIA. [En línea] Octubre de 2005. http://scielo.isciii.es/scielo.php?pid=S0004-06142005000800016&script=sci_arttext&lng=e.
9. **Marín-Blazquez, Antonio Albarracín. 2005.** SUTURAS MECANICAS EN CIRUGIA . [En línea] 2005. <http://scgd3murcia.iespana.es/scgd3murcia/SUTURAS%20MECANICAS.htm>.
10. **Martín Doñate, Cristina. 2004.** *INTERFACES HÁPTICOS. APLICACION EN ENTORNOS VIRTUALES.* 2004.
11. **Robertson, Suzanne y Robertson, James. 2006.** *Mastering the Requirements Process Second Edition.* s.l. : Addison Wesley, 2006.

12. **Wieggers, Karl E. 2006.** *More About Software Requirements.* 2006.
13. —. **2003.** *Software Requirements, Second Edition.* 2003.
14. **Young, Ralph R. 2004 .** *The Requirements Engineering HandBook.* 2004 .

ANEXOS.**Anexo 1: Encuesta realizada a desarrolladores de proyectos de simulación virtual.**

Los Simuladores Virtuales en los que usted ha participado:

1. ¿Han sobrepasado en algún caso el tiempo y costo estimado?

___ Sí.

___ No.

2. En caso de ser afirmativa la respuesta anterior, ¿Qué factores han influido?

___ Falta de comunicación con usuarios y/o clientes.

___ Requisitos y especificación incompletas.

___ Requisitos cambiantes.

___ Falta de apoyo por parte de los directivos.

___ Falta de recursos.

___ Expectativas irreales.

___ Falta de claridad en los objetivos.

___ Estimaciones irreales.

___ Falta de coordinación entre entidades desarrolladoras del proyecto.

___ Consulta de documentos o búsquedas en internet como fuente de información.

___ Otros. ¿Cuáles? _____.

3. ¿Qué nivel de prioridad se le asigna a los requisitos dentro del proceso de desarrollo de software?

___ Alta.

___ Media.

___ Baja.

4. ¿Qué actividades relacionadas con los requisitos acometen durante el proceso de desarrollo de software?

___ Identificar involucrados.

___ Obtención de requisitos.

___ Refinar requisitos.

___ Analizar requisitos.

___ Especificar requisitos.

___ Priorizar requisitos.

- Derivar requisitos.
- Clasificar requisitos.
- Asignar requisitos por subsistema.
- Dar seguimiento a los requisitos.
- Gestionar requisitos.
- Probar y verificar requisitos.
- Validar requisitos.

5. ¿Qué técnica(s) usa para obtener los requisitos de los usuarios y clientes?

- Entrevistas.
- Talleres.
- Tormenta de ideas.
- Cuestionarios.
- Otra(s) ¿Cuál? (es) _____.

6. ¿Qué técnica(s) utiliza para especificar requisitos?

- Casos de Uso.
- Escenarios.
- Pseudocódigo.
- Historias de usuarios.
- Tablas y Árboles de Decisión.
- Modelo Entidad Relación.

7. ¿Qué técnica(s) utiliza para validar los requisitos?

- Listas de chequeo.
- Casos de prueba.
- Construcción de prototipos.
- Otra(s). ¿Cuál? (es) _____.

Anexo 2: Entrevista a clientes y otros interesados en el proyecto.

1. ¿Cómo se realiza el entrenamiento de los médicos cirujanos para una Cirugía de Mínimo Acceso?
2. Para la segunda etapa del SimQ se desarrollarán seis ejercicios: Coordinación, Selección y Cámara Corte, Sutura y Grapado.

2.1. Ejercicio de la Cámara:

- a) ¿Cuáles son los pasos fundamentales que debe realizar el Médico Cirujano para realizar este ejercicio?
- b) ¿Qué habilidades quirúrgicas usted desea medir y evaluar en este ejercicio?
- c) ¿Cuáles serán los criterios de evaluación de estas habilidades?
- d) ¿Qué instrumental quirúrgico se utilizará en este ejercicio?
- e) ¿Qué parámetros se tendrá en cuenta para definir la variación en la complejidad en este ejercicio de un nivel a otro?

2.2. Ejercicio de Coordinación:

- a) ¿Cuáles son los pasos fundamentales que debe realizar el Médico Cirujano para realizar este ejercicio?
- b) ¿Qué habilidades quirúrgicas usted desea medir y evaluar en este ejercicio?
- c) ¿Cuáles serán los criterios de evaluación de estas habilidades?
- d) ¿Qué instrumental quirúrgico se utilizará en este ejercicio?
- e) ¿Qué parámetros se tendrá en cuenta para definir la variación en la complejidad en este ejercicio de un nivel a otro?

2.3. Ejercicio de Selección:

- a) ¿Cuáles son los pasos fundamentales que debe realizar el Médico Cirujano para realizar este ejercicio?

- b)** ¿Qué habilidades quirúrgicas usted desea medir y evaluar en este ejercicio?
- c)** ¿Cuáles serán los criterios de evaluación de estas habilidades?
- d)** ¿Qué instrumental quirúrgico se utilizará en este ejercicio?
- e)** ¿Qué parámetros se tendrá en cuenta para definir la variación en la complejidad en este ejercicio de un nivel a otro?

2.4. Ejercicio de Corte:

- a)** ¿Cuáles son los pasos fundamentales que debe realizar el Médico Cirujano para realizar este ejercicio?
- b)** ¿Qué habilidades quirúrgicas usted desea medir y evaluar en este ejercicio?
- c)** ¿Cuáles serán los criterios de evaluación de estas habilidades?
- d)** ¿Qué instrumental quirúrgico se utilizará en este ejercicio?
- e)** ¿Qué parámetros se tendrá en cuenta para definir la variación en la complejidad en este ejercicio de un nivel a otro?

2.5. Ejercicio de Sutura:

- a)** ¿Cuáles son los pasos fundamentales que debe realizar el Médico Cirujano para realizar este ejercicio?
- b)** ¿Qué habilidades quirúrgicas usted desea medir y evaluar en este ejercicio?
- c)** ¿Cuáles serán los criterios de evaluación de estas habilidades?
- d)** ¿Qué instrumental quirúrgico se utilizará en este ejercicio?
- e)** ¿Qué parámetros se tendrá en cuenta para definir la variación en la complejidad en este ejercicio de un nivel a otro?

2.6. Ejercicio de Grapado:

- a)** ¿Cuáles son los pasos fundamentales que debe realizar el Médico Cirujano para realizar este ejercicio?

- b) ¿Qué habilidades quirúrgicas usted desea medir y evaluar en este ejercicio?
- c) ¿Cuáles serán los criterios de evaluación de estas habilidades?
- d) ¿Qué instrumental quirúrgico se utilizará en este ejercicio?
- e) ¿Qué parámetros se tendrá en cuenta para definir la variación en la complejidad en este ejercicio de un nivel a otro?

Anexo 3: Plantilla para la Descripción Textual de un Caso de Uso.

Caso de Uso:	<Insertar nombre del caso de uso>	
Actores:	<Insertar actor(es) que interviene en el caso de uso>	
Resumen:	<Insertar como y cuando comienza el caso de uso (primera acción a ejecutar)>	
Precondiciones:	<Insertar condición que debe cumplirse para que se inicie el caso de uso (estado inicial)>	
Referencias	<Insertar requisitos asociados al caso de uso>	
Prioridad	<Insertar orden de prioridad del caso de uso>	
Flujo Normal de Eventos		
Acción del Actor	Respuesta del Sistema	
<Insertar acción realizada por el actor>	<Insertar respuesta del sistema>	
Flujo Alterno		
	<Insertar respuesta alternativa del sistema>	
Poscondiciones	<Insertar condición que debe cumplirse para que termine el caso de uso (estado final)>	

Anexo 4: Plantilla para la descripción de un Escenario.

Título: <Insertar título del escenario>
Objetivo: <Insertar objetivo del escenario>
Resumen: <Insertar resumen del escenario>
Precondición: <Insertar precondición>
Actores: <Insertar actor(es) que intervienen en el escenario>
Episodios: <Insertar secuencia de episodios que se suceden en el escenario>
Excepciones: <Insertar excepciones que ocurren dentro del escenario>
<Insertar storyboard y video que permiten visualizar los requisitos involucrados en el escenario>

Anexo 5: Encuesta para valorar la aceptación de la aplicación de las técnicas propuestas para el desarrollo de requisitos en el SimQ con respecto a la opinión del equipo de desarrollo del proyecto.

Teniendo en cuenta el uso de las nuevas técnicas utilizadas para el desarrollo de los requisitos del proyecto responda las siguientes interrogantes.

1. ¿Considera incompleta la especificación de alguno de los requisitos que usted implementó?

___ Sí.

___ No.

2. ¿Tuvo usted que auxiliarse de búsquedas en Internet o consultar documentos por no ser suficiente la especificación de los requisitos dada?

___ Sí.

___ No.

3. ¿En qué medida considera Ud. que se hayan disminuido los cambios en los requerimientos y sus especificaciones por parte del usuario?

___ Alta.

___ Media.

___ Baja.

4. ¿En qué medida se manifestó falta de claridad en los objetivos a lograr en el producto final?

___ Alta.

___ Media.

___ Baja.

5. ¿En qué medida las técnicas de desarrollo de requisitos propuestas mejoró la comunicación con el equipo de desarrollo del proyecto?

___ Alta.

___ Media.

___ Baja.

6. ¿En qué medida considera usted que las técnicas utilizadas facilitaron la implementación de las funcionalidades del SimQ?

___ Alta.

___ Media.

___ Baja.

Anexo 6: Encuesta para valorar los resultados de la aplicación de la nueva propuesta de técnicas para el desarrollo de requisitos en el SimQ con respecto a la posición de los Stakeholder.

1. ¿Considera Ud. que la comunicación con los ingenieros de software con los que interactuó para la realización de este proyecto haya sido buena?

___ Sí.

___ No.

2. En caso de ser positiva su respuesta: ¿En qué medida cree Ud. que las técnicas de desarrollo de requisitos propuestas posibilitaron que se lograra esta adecuada comunicación?

___ Alta.

___ Media.

___ Baja.

3. En qué medida considera Ud. que la propuesta realizada facilitó el buen entendimiento y una alta precisión en las características y funcionalidades definidas para el desarrollo del producto solicitado.

___ Alta.

___ Media.

___ Baja.

4. ¿Considera Ud. que la aplicación de la propuesta brindó la posibilidad de disminuir las solicitudes de cambios en los requerimientos y sus especificaciones?

___ Sí.

___ No.

GLOSARIO DE TÉRMINOS.

Ingeniería de Software (IS): Disciplina o área de la Informática o Ciencia de la Computadora, que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resultan problemas de todos tipos.

Ingeniería de Requisitos (IR): Disciplina que se enmarca dentro de la Ingeniería de software, la cual proponer métodos, técnicas y herramientas que faciliten el trabajo de definición de lo que se quiere de un artefacto de software.

Simulador Quirúrgico (SimQ): Modelos virtuales de distintas partes del cuerpo dotados de herramientas virtuales con las que los cirujanos pueden entrenarse en distintas técnicas quirúrgicas mediante la utilización de entornos virtuales.

Realidad Virtual (RV): Ambiente altamente interactivo donde el usuario participa a través del uso de un computador en un mundo virtualmente real. Es una simulación tridimensional por computadora durante la cual el usuario resulta inmerso tan completamente que esta realidad, de origen artificial, aparenta ser real.

SIMPRO: Centro de Investigación y Desarrollo de Simuladores.

ÍNDICE DE FIGURAS.

Figura 1.1: Hardware de un Simulador Quirúrgico6

Figura 1.2: Pantalla del Simulador GI Mentor con la que entrenan los cirujanos.8

Figura 1.3: Imagen de Intervención a través del simulador LapSim.8

Figura 1.4: Simulador Tower Trainer.9

Figura 1.5: Resultados del informe de GAO.13

Figura 1.6: Resultados del informe de CHAOS.14

Figura 1.7: Origen de los errores de los proyectos de software.16

Figura 1.8: Subcomponentes de la IR.18

Figura 1.9: Procedimiento de desarrollo de requisitos19

Figura 1.10: Procedimiento para la elicitación de requisitos.....20

Figure 1.11: Procedimiento para el análisis de requisitos.23

Figura 1.12: Procedimiento para la especificación de requisitos.24

Figura 1.13: Procedimiento para la validación de requisitos.27

Figura 1.14: Estado de los proyectos de Simulación Virtual.....28

Figura 1.15: Prioridad asignada a los requisitos en proyectos de SV.29

Figura 1.16: Impacto de factores que influyen en el exceso de tiempo y presupuesto en proyectos SV.
.....30

Figura 2.1: Estructura de Desglose del proyecto Simulador Quirúrgico32

Figura 3.1: Interfaz autenticar usuario72

Figura 3.2: Interfaz Principal del Médico Cirujano73

Figura 3.3: Interfaz seleccionar nivel de entrenamiento74

Figure 3.4: Interfaz principal del profesor75

Figura 3.5: Interfaz gestionar evaluaciones76

Figura 3.6: Interfaz principal de la sesión administrador77

Figura 3.7: Interfaz registrar usuario78

Figura 3.8: Interfaz modificar datos del usuario79

Figura 3.9: Eliminar datos del usuario80

Figura 3.10: Factores que reflejan la completitud de la especificación de los requisitos.....82

Figura 3.11: Disminución de los cambios en los requerimientos y sus especificaciones83

Figura 3.12: Comunicación con el equipo de desarrollo83

Figura 3.13: Facilidad en la implementación de las funcionalidades del Simulador a partir de la propuesta84

Figura 3.14: Adecuada comunicación y disminución de las solicitudes de cambio en los requerimientos y sus especificaciones.	85
Figura 3.15: Influencia de la propuesta en el logro de una adecuada comunicación.	85
Figura 3.16: Entendimiento y precisión en las características y funcionalidades definidas.	86

ÍNDICE DE TABLAS.

Tabla 1.1: Origen de los errores de los proyectos de software.	15
Tabla 2.1: Casos de Uso definidos para la primera y segunda etapa de desarrollo del SimQ.	44
Tabla 2.2: Descripción textual del caso de uso Autenticar Usuarios.	45
Tabla 2.3: Descripción textual del caso de uso Gestionar Usuarios del Sistema.	47
Tabla 2.4: Descripción textual del caso de uso Iniciar Ejercicio.	50
Tabla 2.5: Descripción textual del caso de uso Resetear Ejercicio.	55
Tabla 2.6: Descripción textual del caso de uso Emitir Evaluación.	56
Tabla 2.7: Descripción textual del caso de uso Gestionar Evaluación.	57
Tabla 2.8: Escenarios definidos para la primera y segunda etapa de desarrollo del SimQ.	58
Tabla 2.9: Escenario Interactuar con la Cámara.	59
Tabla 2.10: Escenario Focalizar Objeto.	60
Tabla 2.11: Escenario Seleccionar Objeto.	61
Tabla 2.12: Escenario Interactuar con las Pinzas.	63
Tabla 2.13: Escenario Agarrar y Transferir Objeto.	64
Tabla 2.14: Escenario Entrenar Corte.	66
Tabla 2.15: Escenario Entrenar Sutura.	67
Tabla 2.16: Escenario Entrenar Grapado.	69

