

Universidad de las Ciencias Informáticas

Facultad 5 Entornos Virtuales



Estrategia de Aseguramiento de la calidad para el Simulador Quirúrgico.

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

Autores:

Lanny Rivero Canino
Meydi Elena Rodríguez Martínez

Tutores

Lic. Yoisy Pérez Olmos
Ing. Osmanys Valdés Puga

Asesor

Ing. Gerandys Hernández Casanova

Ciudad de la Habana, julio 2008

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

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

(Nombre del Autor)

Meydi E. Rodríguez Martínez

(Nombre del Autor)

Lanny Rivero Canino

(Nombre del Tutor)

Lic. Yoisy Pérez Olmos

(Nombre del Tutor)

Ing. Osmanys Valdés Puga

DATOS DEL CONTACTO

Tutora: Lic. Yoisy Pérez Olmos.

Licenciada en Ciencias de la computación en el año 2005 en la Universidad Central de las Villas con 3 años de experiencia.

E-mail: yoisy@uci.cu

Tutor: Ing. Osmanys Valdés Puga.

Ingeniero en Ciencias Informáticas.

Graduado de la UCI, con un año de experiencia en el tema de la Gráfica Computacional y profesor de un proyecto de Realidad Virtual en la Universidad de las Ciencias Informáticas.

Institución: Universidad de las Ciencias Informáticas (UCI).

Categoría Docente: Profesor Adiestrado

E-mail: ovaldes@uci.cu

Asesor: Ing. Gerandys Hernández Casanova.

Ingeniero en Ciencias Informáticas. Graduado en Julio de 2007. Desde este año trabaja en la Universidad de las Ciencias Informáticas como profesor de Administración de Empresa y Contabilidad y Finanzas. Se desempeña actualmente como líder del proyecto Calidad de la Facultad 5.

E-mail: ghernandez@uci.cu

Dedicatoria

A mi abuelito que a pesar de no estar a mi lado en estos últimos años ha sido mi fuente de inspiración...

A mi mamá por ser mi mejor amiga... a mi abuela por ser mi razón de existir... a las dos por todos sus sacrificios para que yo terminara mi carrera y por estar siempre presentes en los momentos más difíciles... a ellas les debo lo que soy hoy...

A mi esposo por todo el amor y apoyo incondicional que me ha brindado durante todo este tiempo, por su paciencia cada vez que tenía un problema y ayudarme a entender que si se puede...

A ellos les dedico mi tesis...su sueño...mi sueño.

Meydi

A mis padres que siempre están y estarán en mis pensamientos, por ser mi máxima fuente de inspiración, por darme vida y todo su amor, por formarme como la persona que soy hoy.

A mi mamá Lourdes, por haber sido madre, amiga, ternura y pasión, por todo su apoyo incondicional en todos los momentos difíciles por los cuales pase.

A mi papá, Oscar, por el sacrificio de tantos años y por haber depositado en mí toda la confianza del mundo.

A Cecilia por preocuparse tanto por mi y ayudarme siempre que lo necesitaba, por su apoyo ilimitado.

A mis hermanos Oscaritin y Vianney por ser mis hermanos y quererme tanto.

A mi novio por tanto amor y dedicación.

A todos dedico mi tesis...mis esfuerzos y mis sueños.

Lanny

Agradecimientos Compartidos

A la Revolución Cubana y a nuestro eterno Comandante en Jefe Fidel Castro por brindarnos la oportunidad de haber estudiado en la primera universidad surgida en el calor de la batalla de ideas.

A la Universidad de las Ciencias Informáticas por darnos la oportunidad de estudiar en esta magnífica escuela y de utilizar los medios y condiciones necesarios para llevar a cabo esta investigación.

A la Decana de nuestra facultad Mayra por ser tan comprensiva y tener tanta paciencia.

A profesores como Gerandys, Pepe, Jandrich y Julián por ayudarnos y aportarnos ideas para seguir adelante.

A nuestros tutores Yoisy y Osmanys gracias por su apoyo durante el desarrollo del trabajo.

A nuestros compañeros de aula por haber compartido estos cinco años de nuevas experiencias.

A todas aquellas personas que durante estos años fueron compañeros, amigos y que de una forma u otra pusieron un granito de arena para que este sueño se hiciera realidad, gracias a todos por hacernos reír y pasar buenos momentos, por hacernos la estancia en la universidad más agradable y hacer posible la realización de este trabajo.

Meydi y Lanny

AGRADECIMIENTOS

Sería difícil agradecerles a todas las personas que de una forma u otra me ayudaron durante toda la carrera e hicieron posible que llegara hasta aquí.

Muchas gracias a mi mamá por creer en mí, por su amor durante toda mi vida y por darme fuerzas para seguir adelante cada día.

A mi abuelita por su dedicación y esfuerzo para estar a mi lado en todo momento, y por ayudar a mi mamá a formarme en la persona que hoy soy.

A mis tíos que fueron mi apoyo en los primeros años de mi carrera cuando me encontraba sola y lejos de mi casa.

A mi esposo que desde que lo conocí me ayudó a superar los problemas y me dio aliento para seguir adelante, conocerte ha sido el mejor regalo que me ha dado la vida.

A mi suegra que desde el primer momento me acogió como su hija y me ofreció todo su ternura y comprensión.

A mi segundos abuelitos, Mérida y Humberto por brindarme todo su amor.

A Yure que más que mi amiga es mi hermana y ha estado a mi lado para darme consejos cada vez que los necesité.

A mi amiga y compañera de tesis Lanny que me ayudó a ser fuerte y sin ella este sueño no se hubiera realizado.

A mi amigo Rigo que sin él no hubiera vencido asignaturas como programación.

Al profesor Fermín que confió en mí y me apoyó en situaciones difíciles.

A mis amigos por los ratos compartidos y estar presente en los buenos y malos momentos.

Meydi

Agradecimientos

En primer lugar quiero agradecerles a mis padres por ser parte de mi vida en todo momento, por brindarme su amor, su apoyo y cariño en cada momento de mi vida, por estar a mi lado y darme fuerzas para ser cada día mejor.

-A mi mami a quien no tengo forma de agradecerle por todo lo que me ha dado, por ser mi sostén, mi guía, mi amiga, sin ti no hubiera sido capaz de convertirme en la persona que soy y tratar de ser mejor, gracias por darme valor, ayudarme a ser fuerte y responsable, por depositar tu confianza en mi y por estar junto a mi en todos los momentos, mami gracias por existir porque sin ti mi vida no tendría sentido.

-A mi papi por darme todo tu amor y ternura, por exigirme que fuera una persona preparada y superarme profesionalmente para la vida porque me sirvió para ser lo que hoy soy, por confiar en mí y por ser alguien importante para mi.

-A mis dos hermanos que adoro, gracias por preocuparse por mí, por ser mi sostén y apoyo en mi alegría.

-A Cecilia que tanto me ha apoyado en mi vida personal y profesional, por su confianza y ayuda ilimitada... Por poder contar contigo cada vez que lo necesitaba... Por todos tus consejos.

A Leandro, mi novio, mi compañero y mucho más que eso, por comprenderme, aguantarme, y estar siempre a mi lado, eres de las mejores cosas que me han pasado en la vida.

-A sus padres Maritza y Anibal, y a Laro que siempre estuvieron cerca en representación de mi familia.

-Un agradecimiento especial a mi compañera de tesis, Meydi, gracias por ayudarme cuando me hizo falta, por las traspasadas y gracias por confiar en mí para hacer esta tesis. Saludos especiales a sus familiares.

-A todos quiero agradecerle desde lo más profundo de mi corazón..... Gracias

Lanny

RESUMEN

El presente trabajo que lleva por título “Estrategia de Aseguramiento de la Calidad para el Simulador Quirúrgico” cuenta con un estudio del estado del arte enmarcado en elementos relacionados con la calidad del software, puntualizándose temas como el aseguramiento y estándares de calidad, además de pruebas y métricas de software. Se realizó también una descripción detallada de la situación actual del aseguramiento de la calidad en el proyecto Simulador Quirúrgico y finalmente se especifican los principales señalamientos detectados en el proyecto en una auditoría realizada por la dirección de calidad del software de la UCI.

Todo esto se realiza con el objetivo de definir una estrategia de aseguramiento de la calidad para el proyecto Simulador Quirúrgico que posibilite lograr un nivel de calidad en los productos a obtener. Esta propuesta está sustentada por el Plan de aseguramiento de la calidad como artefacto fundamental, además de la definición del equipo de calidad, y los procesos de revisiones y pruebas en el proyecto. Finalmente se realizó la validación de la propuesta a partir del uso del método Delphi obteniéndose resultados satisfactorios que aprueban la aplicación de la propuesta a partir de la segunda etapa de desarrollo del proyecto.

PALABRAS CLAVES

Simulador Quirúrgico, Calidad de Software, Aseguramiento de la Calidad,

Plan de aseguramiento de la calidad.

ÍNDICE

AGRADECIMIENTOS	5
RESUMEN.....	7
INTRODUCCIÓN.....	11
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	15
INTRODUCCIÓN.....	15
1.1. SURGIMIENTO DE LOS SIMULADORES QUIRÚRGICOS.....	15
1.2. ¿QUÉ ES CALIDAD?.....	16
1.2.1 ¿Qué es calidad del Software?	17
1.2.2 Calidad a nivel del producto.....	17
1.2.3 Factores que determinan la calidad del software.	18
1.2.4 Calidad del software en Cuba y en la Universidad de las Ciencias Informáticas (UCI).	19
1.3 GESTIÓN DE LA CALIDAD.....	21
1.3.1 Aseguramiento de la Calidad o Garantía de la Calidad.	21
1.3.2 Revisiones de Software.	23
1.3.3 Control de la calidad del software.....	25
1.4 PRUEBAS.....	26
1.5 MÉTRICAS DE SOFTWARE	27
1.6 METODOLOGÍA RUP.....	28
1.6.1 Ciclo de Vida de RUP.	28
1.6.2 Fases e Hitos de RUP.	29
1.6.3 Definición de Roles según la metodología RUP.....	30
1.7 ESTÁNDARES DE CALIDAD.	30
1.7.1 CMMI (Modelo de Madurez de Capacidad Integrado).....	31
1.7.2 Estándar ISO 9000-3.....	32
CAPÍTULO 2: DESCRIPCIÓN Y SITUACIÓN ACTUAL DEL PROYECTO SIMULADOR QUIRÚRGICO.....	35
INTRODUCCIÓN.....	35
2.1 DESCRIPCIÓN DEL SIMULADOR QUIRÚRGICO.....	35

2.2 HERRAMIENTAS Y CONDICIONES DE DESARROLLO EN EL PROYECTO PRODUCTIVO SIMULADOR QUIRÚRGICO.....	37
2.3 PROCESO DE DESARROLLO EN EL SIMULADOR QUIRÚRGICO.....	38
2.4 DESCRIPCIÓN DEL PROCESO DE ASEGURAMIENTO DE LA CALIDAD EN EL SIMULADOR QUIRÚRGICO.....	40
2.5 RESULTADOS DE AUDITORÍA REALIZADA AL PROYECTO SIMULADOR QUIRÚRGICO... ..	46
CAPÍTULO 3: PROPUESTA Y DESCRIPCIÓN DE LA SOLUCIÓN.....	48
INTRODUCCIÓN.....	48
3.1 CARACTERÍSTICAS DE LA ESTRATEGIA.....	48
3.2 DEFINICIÓN DEL EQUIPO DE ASEGURAMIENTO DE LA CALIDAD EN EL SIMULADOR QUIRÚRGICO.....	49
3.3 DEFINICIÓN DEL PLAN DE ASEGURAMIENTO DE LA CALIDAD DEL SOFTWARE EN EL SIMULADOR QUIRÚRGICO.....	50
3.3.1 Objetivos de calidad para el proyecto.....	51
3.3.2 Métrica para el proceso de revisiones de software en el Simulador Quirúrgico	52
3.3.3 Estándares y guías.....	53
3.3.4 Tareas generales de revisiones.....	53
3.3.5 Pruebas.....	57
3.3.6 Herramientas, técnicas y metodologías.....	59
3.3.7 Resolución de problema y acción correctiva.....	60
3.3.8 Registros de calidad.....	60
3.4 PROCESO DE REVISIONES EN EL SIMULADOR QUIRÚRGICO.....	61
3.5 PROCESO DE PRUEBAS EN EL SIMULADOR QUIRÚRGICO.....	62
3.6 VALIDACIÓN DE LA ESTRATEGIA.....	63
CONCLUSIONES.....	68
RECOMENDACIONES.....	69
REFERENCIA BIBLIOGRÁFICA.....	70
ANEXOS.....	73
ANEXO1 FACTORES MACKALL.....	73

ANEXO 2 FASES, ITERACIONES Y FLUJOS DE TRABAJO DE RUP.....	73
ANEXO 3 FASES E HITOS.....	74
ANEXO 4 ÁREAS DE DESARROLLO DEL MODELO CMMI.....	74
ANEXO 5 ESTÁNDARES Y NORMAS UTILIZADOS.....	75
ANEXO 6 LISTA DE CHEQUEO PARA LA REVISIÓN ADMINISTRATIVA DEL SIMULADOR QUIRÚRGICO.....	78
ANEXO 7 LISTA DE CHEQUEO PARA LA REVISIÓN DE LOS REQUISITOS EN EL SIMULADOR QUIRÚRGICO.....	79
ANEXO 8 LISTA DE CHEQUEO PARA LA REVISIÓN DE LA ARQUITECTURA EN EL SIMULADOR QUIRÚRGICO.....	80
ANEXO 9 LISTA DE CHEQUEO PARA LA REVISIÓN AL DISEÑO EN EL SIMULADOR QUIRÚRGICO.....	82
ANEXO 10 LISTA DE CHEQUEO PARA LA REVISIÓN A LA DOCUMENTACIÓN.....	83
ANEXO 11 LISTA DE CHEQUEO PARA LA REVISIÓN A LA GESTIÓN DE CONFIGURACIÓN EN EL SIMULADOR QUIRÚRGICO.....	85
ANEXO 12 REGISTRO DE NO CONFORMIDADES.....	88
ANEXO 13 REGISTRO DE RESPUESTA A LAS NO CONFORMIDADES.....	89
ANEXO 14 ENCUESTA REALIZADA PARA LA VALIDACIÓN.....	89
GLOSARIO DE TÉRMINOS Y SIGLAS.....	91

INTRODUCCIÓN.

A nivel mundial la industria del software goza de una demanda tan elevada que ha llegado a convertirse en un enorme gigante que se desarrolla a ritmo acelerado. Todas las empresas desean producir software de alta calidad que obedezca las exigencias y responda a la planificación en el menor tiempo posible y a costos mínimos para brindarle al cliente un producto con calidad. Este es uno de los vitales desafíos a lograr al construir un producto con calidad.

Desde aproximadamente la década del 70 hasta el día de hoy la calidad del software es impulso de inquietud para todos aquellos relacionados con la creación de software. La calidad es sinónimo de flexibilidad, eficiencia, seguridad, confiabilidad, corrección, integridad, portabilidad y usabilidad. La calidad del software varía de un producto a otro y puede ser medida después de elaborado el producto, pero esto implicaría un alto costo, pues se pueden descubrir inconvenientes. Por lo que es ineludible tener en cuenta tanto la obtención de la calidad como su control durante todo el período de vida del software (Carrasco, 2004).

El software adquiere cada vez mayor relevancia en las actividades que se realizan y en los dispositivos que se manejan para ello, estando así presente en muchos aspectos de la vida cotidiana: en el banco, transporte, educación, entretenimiento, medicina, etc.; pero es necesario que esté construido con calidad, por lo que se precisa, entre otras cosas, asegurar la calidad del software.

Nuestro país ejecuta algunas inversiones en la industria del software, ya que informatizar a la sociedad y convertir esta industria en una de las ramas más productivas y aportadoras de recursos para el estado son objetivos principales determinados por el gobierno para los próximos años. De ahí que se haga necesaria la elaboración de productos que admitan colocar el software en el mercado mundial. En medio de ese furor informático y de la batalla de ideas que libra nuestro país surge la Universidad de las Ciencias Informáticas, la cual tiene como primordial misión cambiar el destino de la industria cubana del software dada la unión de miles de especialistas y recursos para su fabricación. En ella se realizan importantes proyectos productivos,

alcanzados por estudiantes y profesores de las distintas facultades. Un ejemplo es la producción de un Simulador Quirúrgico que tiene como objetivo fundamental entrenar a los médicos cirujanos que practican la Cirugía de Mínimo Acceso (CMA), minimizando los errores asociados a un adiestramiento insuficiente y marcando un cambio en los conceptos de aprendizaje y enseñanza de estas técnicas a partir del empleo de la realidad virtual. De esta manera se logra el entrenamiento de un mayor número de educandos en un menor tiempo y espacio con costos de mantenimiento mínimos, brindando la posibilidad además, de evaluar objetivamente y medir el nivel de competencia adquirido por los alumnos.

Cualquier proyecto de software se empeña en obtener un producto de calidad, por lo que es inevitable el manejo de estrategias y procedimientos llevando consigo el aseguramiento de la calidad del proyecto que tiene como propósito aportar protección en cada fase del proceso de desarrollo.

El Simulador Quirúrgico se encuentra en la primera etapa de vida, por lo que carece de los mecanismos necesarios para el aseguramiento de la calidad interna del software en el período de desarrollo y por consiguiente de la calidad en la entrega del producto final. Por tal motivo se hace indispensable la realización de una estrategia de aseguramiento de la calidad con el objetivo de obtener un producto de calidad y aceptación. Para lograr la calidad requerida resulta preciso definir un conjunto de actividades para reducir, eliminar y prevenir las deficiencias de calidad del producto a obtener y alcanzar una razonable confianza en que las prestaciones y servicios esperados por el cliente o el usuario queden satisfechas.

A partir de la situación expuesta anteriormente se define como **problema científico** de la presente investigación ¿Cómo asegurar que los procesos y productos resultantes del proyecto Simulador Quirúrgico estén de acuerdo a los planes establecidos?

El **objeto de estudio** de la investigación se basa en el proceso de desarrollo de software del Simulador Quirúrgico y se concreta como **campo de acción** el proceso de aseguramiento de la calidad en el Simulador Quirúrgico.

Para dar solución al problema planteado se define el siguiente **objetivo general** de la investigación: Definir una estrategia de aseguramiento de la calidad en el proyecto Simulador Quirúrgico.

Para cumplimentar el objetivo general se definen a continuación algunos **objetivos específicos**:

- Realizar un estudio de los diferentes enfoques de aseguramiento de la calidad según algunos de los modelos, estándares y normas para el desarrollo del software.
- Describir el proceso de aseguramiento de la calidad en el Simulador Quirúrgico actualmente.
- Proponer una estrategia de aseguramiento de la calidad en el Simulador Quirúrgico.
- Validar la estrategia definida a partir de la aplicación del método Delphi.

Seguidamente se citan los métodos científicos utilizados para el desarrollo de la investigación:

Métodos Teóricos

Histórico-Lógico: Este método ayudará a comprender el desarrollo, la historia y la lógica del fenómeno analizado.

Analítico-Sintético: Este método permitirá identificar de toda la información que aborda el tema estudiado los elementos más importantes y de mayor utilidad para esta investigación.

Como principal **aporte práctico** del trabajo se obtendrá una estrategia de aseguramiento de la calidad para el proyecto Simulador Quirúrgico que posibilite la entrega al cliente de un producto con el máximo nivel de calidad y acorde a sus requerimientos.

El trabajo está estructurado en 3 capítulos:

Capítulo 1: Fundamentación Teórica: se realizará una investigación sobre los simuladores quirúrgicos y posteriormente se especificará el término calidad, cómo se puede asegurar y cuáles son algunos de los estándares de calidad existentes.

Capítulo 2: Descripción y situación actual: se describe el proyecto Simulador Quirúrgico y la situación actual en que se encuentra haciendo énfasis en su proceso de aseguramiento de la calidad. Se especifica además, los resultados obtenidos en el proyecto en una auditoría realizada por la dirección de calidad de la UCI.

Capítulo 3: Propuesta y descripción de la solución: se realiza la propuesta donde se definen todas las tareas que se deben ejecutar para que el producto final satisfaga las necesidades del cliente, además de validar dicha estrategia a partir de los criterios de algunos expertos utilizando el método Dephi.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Introducción

En el presente capítulo se materializan diferentes aspectos entre los que se encuentran: calidad, gestión de la calidad, calidad en Cuba y en la UCI, diferentes estándares como son CMMI e ISO, así como el enfoque que ellos muestran sobre el aseguramiento de la calidad, además de las revisiones de software; ofreciendo una serie de conceptos y definiciones que fundamentan teóricamente la investigación.

1.1. Surgimiento de los Simuladores Quirúrgicos.

Un simulador no es más que un aparato que reproduce el comportamiento de un sistema en determinadas condiciones, aplicado generalmente para el entrenamiento de quienes deben manejar dicho sistema. Los simuladores reproducen sensaciones que en realidad no están sucediendo (Espasa, 2005).

En el caso de la medicina se han desarrollado simuladores quirúrgicos para posibilitarle al cirujano el entrenamiento y la planificación de intervenciones quirúrgicas en escenas virtuales, disminuyendo los riesgos ya que le proporciona al cirujano la interacción con dicha escena. Los simuladores quirúrgicos son modelos virtuales que los cirujanos utilizan para adiestrarse en las diferentes habilidades médicas mediante el uso de ambientes virtuales. Estos simuladores requieren de la respuesta realista y en tiempo de los objetos y órganos del paciente, que sean capaces de deformarse ante las acciones típicas quirúrgicas como corte, sutura o cauterización. Mundialmente se han desarrollado varios de ellos que se utilizan en la incursión de cirugías tanto laparoscópicas como endoscópicas, ejemplos: el ProMIS, GI Mentor y LapSim System.

Para lograr que los Simuladores Quirúrgicos tengan un funcionamiento óptimo es necesario que alcancen la calidad requerida tanto en el proceso de desarrollo como en el producto final porque cada vez más los errores en el software repercuten directa o indirectamente en graves consecuencias para la organización.

1.2. ¿Qué es Calidad?

El término “calidad” es un concepto escurridizo, fácil de visualizar y sin embargo difícil de medir, a continuación se encuentran algunas de las definiciones dadas por los padres de la idea de la gestión de la calidad.

Según *Dr. Joseph M. Juran*, calidad es “el conjunto de características de un producto que satisfacen las necesidades de los clientes y en consecuencia hacen satisfactorio el producto” (Juran, 1979).

W. Edwards Deming, expresa que calidad es “Conformidad con los requisitos y confianza en el funcionamiento” (Deming, 1989).

También definieron calidad los estándares internacionales, por ejemplo:

CMMI define calidad como la capacidad de un conjunto de características inherentes de un producto, componente de producto o proceso para satisfacer los requisitos de los clientes (CMMI, 2006).

La IEEE en 1990 puntualiza que la calidad es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario (IEEE, 1990).

La norma ISO 9000:2000 la especifica como el grado en el que un conjunto de características inherentes cumple con los requisitos (ISO 9000, 2000).

Para las autoras de este trabajo, teniendo presente todas las definiciones inicialmente señaladas, la calidad es algo relativo ya que depende de los requisitos o necesidades que se desean satisfacer, concediendo la capacidad al producto de cumplir con las exigencias del cliente y cumpliendo con los requerimientos establecidos de forma útil y económica.

1.2.1 ¿Qué es calidad del Software?

Debido a la importancia de la calidad del software, en los estudios realizados distintos autores la han descrito de varias maneras:

La Calidad del Software es el conjunto de cualidades que la caracterizan y que determinan su utilidad y existencia, plantea un adecuado balanceo de eficiencia, confiabilidad, facilidad de mantenimiento, portabilidad, facilidad de uso, seguridad e integridad (Scalone, 2006).

También Pressman define la calidad del software como la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente (Pressman, 2005).

La calidad del software puede medirse después de elaborado el producto. Pero esto puede resultar muy costoso si se detectan problemas derivados de imperfecciones en el diseño, por lo que es imprescindible tener en cuenta tanto la obtención de la calidad como su control durante todas las etapas del ciclo de vida del software (Ferré, 2005).

Después de analizar las diferentes definiciones se puede decir que el sostén de las medidas de calidad es cumplir con los requisitos precisados, porque con una falta de relación entre el software y los requerimientos basta para que exista un error de calidad. Además se enfatiza que se debe seguir una metodología ya que sirven de guía para el desarrollo del software basándose en un conjunto de criterios de desarrollo definidos y de no ser así el producto no tendrá la calidad requerida.

1.2.2 Calidad a nivel del producto.

La calidad la establece esencialmente el cliente, y se procura que el diseño y la fabricación del producto para la venta satisfagan estos requerimientos (Juran, 1993).

La calidad del producto no se debe medir al final sin haberla medido durante todo el proceso de desarrollo. Medir la calidad de un producto final ayuda a solidificar todo el

proceso llevado a cabo, detecta los errores que puedan quedar y daría el punto final a un producto de excelencia. El objetivo de la calidad no es lograr un producto perfecto, sino obtener un producto que cumpla con las características necesarias y suficientes que precisa el cliente, de ahí la necesidad de conocer con detalle las expectativas reales del usuario.

La calidad del producto de software abarca los siguientes aspectos: (Scalone, 2006)

- *Calidad Interna*: medible a partir de las características intrínsecas, como el código fuente.
- *Calidad Externa*: medible en el comportamiento del producto, como en una prueba.
- *Calidad en Uso*: durante la utilización efectiva por parte del usuario.

El objetivo no es ver estos aspectos como lo suficientemente profundos y recogedores de procesos, capaces de fundamentar la solución perfecta para obtener un producto con calidad, no serían suficientes para satisfacer a los clientes actuales. Estos elementos serían un complemento que ayudaría en gran medida a mejorar la calidad del producto.

El desarrollo de productos software es una actividad sujeta a muchos factores que la pueden hacer poco confiable. Esto se debe a que la calidad del producto depende de agentes que en ocasiones sufren varios cambios. Un error que se cometa puede afectar todo un proceso.

1.2.3 Factores que determinan la calidad del software.

Los factores que determinan la calidad del software se clasifican en tres grupos, estos son los siguientes: (*Ver anexo 1*)

Operaciones del producto (características operativas):

- *Corrección*: Hasta qué punto un programa cumple sus especificaciones y satisface los objetivos del usuario.

- **Fiabilidad:** Hasta qué punto se puede confiar en el funcionamiento sin errores del programa.
- **Eficiencia:** Cantidad de código y de recursos informáticos (CPU, memoria) que precisa un programa para desempeñar su función.
- **Integridad:** Hasta qué punto se controlan los accesos ilegales a programas o datos por personas no autorizadas.
- **Facilidad de uso:** El coste y esfuerzo de aprender a manejar un producto, preparar la entrada de datos e interpretar la salida del mismo.

Revisión del producto (capacidad para soportar cambios):

- **Facilidad de mantenimiento:** El coste de localizar y corregir defectos en un programa que aparecen durante su funcionamiento.
- **Facilidad de prueba:** El coste de probar un programa para comprobar que satisface sus requisitos.
- **Flexibilidad:** El coste de modificación del producto cuando cambian sus especificaciones.

Transición del producto (adaptabilidad a nuevos entornos):

- **Portabilidad (Transportabilidad):** El coste de transportar o migrar un producto de una configuración hardware o entorno operativo a otro.
- **Facilidad de Reutilización:** Hasta qué punto se puede transferir un módulo o programa del presente sistema a otra aplicación, y con qué esfuerzo.
- **Interoperabilidad:** El coste y esfuerzo necesario para hacer que el software pueda operar conjuntamente con otros sistemas o aplicaciones software externos (Mendoza, 2005).

1.2.4 Calidad del software en Cuba y en la Universidad de las Ciencias Informáticas (UCI).

El desarrollo de una Industria Nacional de Software es una labor de gran importancia para el estado cubano debido a la perspectiva económica que posee (Baeza, 1995), ya

Capítulo I: Fundamentación Teórica

que garantiza un grupo importante de actividades del país. Pero aún así, los resultados obtenidos no satisfacen las expectativas, la productividad es baja, es impredecible la cantidad real de recursos a consumir en tiempo, el trabajo realizado la mayoría de las veces carece de la calidad y profesionalidad requerida, además que los proyectos no son entregados en el tiempo establecido (Febles, 2000).

Los centros de educación superior siempre han apoyado el desarrollo de las técnicas para lograr la calidad del software en el país, un ejemplo es el Instituto Central de Investigaciones Digitales (ICID). La empresa de Producción y Desarrollo de Software y Calidad (SOFTCAL) en los años 90 desarrolló un sistema de documentación y otro de control de calidad nacional. También se encuentran empresas como son Desoft y Softel que tienen poca experiencia y carecen de una estrategia de aseguramiento de la calidad trayendo consigo que los procesos sean inestables e inmaduros.

La creación de la Universidad de las Ciencias Informáticas (UCI) y su concepción productiva ha dado un vuelco en los últimos años al tema de la calidad del software pues se ha ideado como agente de cambio en la industria cubana del software debido a la concentración de miles de especialistas y recursos para la producción del software.

Uno de los momentos claves en el desarrollo por la UCI del tema ha sido la creación de la Dirección Calidad de Software de la UCI, encargado de controlar y dirigir la calidad de los productos desarrollados en la universidad, además de su proceso de producción a nivel general, así como de definir los lineamientos mínimos de calidad. Se ha conseguido establecer un conjunto de documentos y plantillas que contribuyen a la organización de los proyectos conformando el Expediente del Proyecto donde quedan registradas las actividades, artefactos y procesos realizados en el transcurso del proyecto. Cada facultad conforma un grupo de calidad los que están facultados para controlar la calidad de sus proyectos y son dirigidos por los asesores de calidad de cada una respectivamente.

1.3 Gestión de la calidad.

La Calidad se logra a través de la Gestión de la Calidad, la cual según ISO 9000:2000, consiste en la realización de actividades coordinadas que permiten dirigir y controlar una organización en lo relativo a la calidad (ISO 9000, 2000).

De igual forma se dice que la gestión de la calidad es un aspecto de la función de gestión que determina y aplica la política de la calidad, los objetivos y las responsabilidades y que lo realiza con medios tales como la planificación de la calidad, el control de la calidad, el aseguramiento de la calidad y la mejora de la calidad. La gestión de la calidad es responsabilidad de todos los niveles ejecutivos, pero debe estar guiada por la alta dirección. Su realización involucra a todos los miembros de la organización. Para gestionar la calidad se realizan dos actividades fundamentales, el control de la calidad y el aseguramiento de la calidad (Toiran, 2006).

CMMI plantea que un sistema de gestión de la calidad es el conjunto de elementos interrelacionados de una empresa u organización, por los cuales se administra de forma planificada la calidad, en la búsqueda de la satisfacción de su cliente. Entre dichos elementos, los principales son: la estructura de la organización, sus procesos, sus documentos y sus recursos (CMMI, 2006).

1.3.1 Aseguramiento de la Calidad o Garantía de la Calidad.

El aseguramiento de la calidad contiene todas aquellas actividades que se ejecutan con el objetivo de asegurar un nivel de calidad en el producto desarrollado realizándose de forma independiente al equipo de desarrollo. (Pressman, 2005)

Según la Norma ISO 9000:2000, el aseguramiento de la calidad es la parte de la gestión de la calidad orientada a proporcionar confianza en que se cumplirán los requisitos de calidad. (ISO 9000, 2000)

El aseguramiento de la calidad es una actividad de protección, que se utiliza durante todo el proceso de desarrollo del software. Comprende procedimientos para una mejor aplicación de métodos y herramientas, revisiones técnicas formales, técnicas y

estrategias de pruebas, procedimientos de garantía de ajuste a los estándares y mecanismos de medida e información. Y por supuesto, incluye la garantía de que un sistema cumpla las especificaciones y los requisitos para el uso y desempeño deseado. (Dapena, 2005)

El objetivo del aseguramiento de la calidad es facilitar la gestión para comunicar los informes necesarios referentes a la calidad del producto, por lo que se va consiguiendo una visión más amplia y segura de que la calidad del producto está cumpliendo sus objetivos. Si los datos proporcionados a través del aseguramiento de la calidad identifican problemas, es responsabilidad de la gestión enfrentar los problemas y utilizar los recursos necesarios para solucionar aspectos de calidad. (Pressman, 2005)

Después de haber analizado todos estos criterios sobre cómo asegurar la calidad de un proyecto las autoras de este trabajo consideran que el aseguramiento de la calidad se debe planificar desde el inicio, pues está basado en la prevención y es más productivo prevenir los fallos de calidad que modificarlos. Además existe la necesidad de seleccionar y aplicar una misma forma de controlar el desarrollo del software para todas las etapas de vida de un proyecto.

Una vez que se comprende lo que es calidad un equipo de software debe identificar un conjunto de actividades de aseguramiento de la calidad del software que eliminarán los errores de los productos realizados antes de que ocurran. Estas actividades son : (Pressman, 2005)

- Establecimiento de un plan de aseguramiento de la calidad para el proyecto. El plan se elabora durante la planificación del proyecto y es inspeccionado por todas las partes interesadas.
- Participación en el desarrollo de la descripción del proceso de software del proyecto. El grupo de ingeniería de software elige el proceso a utilizar en el trabajo que se va a efectuar. El equipo de aseguramiento de la calidad examina la descripción del proceso para ajustarse a las políticas de la empresa, los estándares internos del software, los estándares impuestos externamente, y otras partes del plan del proyecto del software.

- Revisión de las actividades de ingeniería del software para comprobar su ajuste al proceso de software definido. El equipo de aseguramiento de la calidad identifica, documenta y sigue la pista de las desviaciones desde el proceso y verifica que se han hecho las correcciones.
- Auditorías de los productos de software escogidos para verificar el ajuste con los definidos como parte del proceso del software. El equipo de aseguramiento de la calidad revisa los productos seleccionados, identifica, documenta y sigue la pista de las desviaciones, verifica que se han hecho las correcciones, e informa periódicamente de los resultados de su trabajo al gestor del proyecto.
- Asegurar que las desviaciones del trabajo y los productos del software se documentan y se manejan de acuerdo con un procedimiento establecido. Las desviaciones se pueden encontrar en el plan del proyecto, en la descripción del proceso, en los estándares aplicables o en los productos técnicos.
- Registrar lo que no se ajuste a los requisitos y comunicar a sus superiores. Los elementos que no se ajustan a los requisitos están bajo seguimiento hasta que se resuelven.

El aseguramiento de la calidad del software se encuentra vigente en actividades como:

- En el enfoque de gestión de calidad.
- Tecnología de Ingeniería de Software efectiva (métodos y herramientas).
- Revisiones Técnicas Formales que se aplican durante el proceso del software.
- Una estrategia de prueba multiescala.
- Un control de la documentación del software y de los cambios realizados.
- Un procedimiento que asegure un ajuste a los estándares de desarrollo del software.

1.3.2 Revisiones de Software.

Las revisiones de software son un "filtro" para el proceso de Ingeniería del Software y son aplicadas en varios momentos del desarrollo del software. Sirven para detectar errores y defectos que pueden ser eliminados, para "purificar" las actividades de la Ingeniería del Software que suceden como resultado del análisis, diseño y codificación,

redundan en una mejora de la calidad del objeto que se examina, provoca indirectamente una mejora de la calidad del proceso de desarrollo al facilitar la comunicación entre los miembros del equipo de desarrollo, además facilitan el control del coste y el tiempo. (Scalone, 2006)

Dentro de las revisiones se encuentran las Inspecciones, que es donde los participantes van leyendo el documento, paso a paso, guiados por el autor, comprobando en cada paso el cumplimiento de los criterios de una **Lista de Chequeo** (Checklist) o Lista de comprobación como también se le conoce.

La lista de chequeo es una técnica que se utiliza para realizar las inspecciones, 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 la propia lista de chequeo. Esta lista será cumplimentada por el grupo de personas entrevistadas o simplemente para recoger información en forma independiente de una entrevista (Escalona, 2002). Al intentar dar respuesta a las preguntas formuladas deben salir a la luz los problemas que puedan existir. Cada revisor debe completar la lista y anotar cualquier tipo de pregunta o defecto detectado (Antonio, 2002).

Existen diferentes tipos de revisiones de software, una de las más importantes es la **Revisión Técnica Formal** (RTF). Es el filtro más efectivo desde el punto de vista del aseguramiento de la calidad y es un medio eficaz para mejorar la calidad del software. (Scalone, 2006)

El objetivo principal de las RTF es encontrar errores durante el proceso, de forma que no se conviertan en defectos después de la entrega del software. El beneficio de las RTF es el descubrimiento de errores al principio para que no se propaguen al paso siguiente del proceso de software, otros de sus objetivos son: (Scalone, 2006)

- Descubrir errores en la función, la lógica o la implementación de cualquier representación del software.
- Verificar que el software bajo la revisión alcance sus requisitos.

- Asegurar que el software haya sido representado de acuerdo con ciertos estándares predefinidos.
- Obtener un software desarrollado de forma uniforme.
- Hacer que los proyectos sean más manejables.

Las Revisiones Técnicas Formales permiten establecer un marco común para la definición de distintas etapas de revisión en el ciclo de vida del software, este mecanismo no solo está pensado para las etapas tempranas del ciclo de vida, sino que puede y debe ser utilizado en etapas como la de prueba de software y mantenimiento. El mecanismo más común para su implementación es la reunión de revisión, la cual deberá regirse, para asegurar su éxito por una buena planificación, control y sobre todo, por la participación dedicada de todos y cada uno de los involucrados. (Control, 2000)

1.3.3 Control de la calidad del software.

“El control de la calidad es el conjunto de técnicas y actividades, de carácter operativo, utilizadas para verificar los requisitos relativos a la calidad del producto o servicio” (Torregrosa).

El control de la calidad es una sucesión de inspecciones y pruebas realizadas en el transcurso del proceso del software para cerciorarse de que el producto cumple con los requerimientos establecidos. Cuenta con un bucle de realimentación conocido como el feedback del proceso que concibió al producto. La realimentación y la medición fusionadas posibilitan la mejora del proceso cuando existe una falla de los productos de trabajos elaborados al efectuar sus especificaciones. Este enfoque ve el control de la calidad como parte del proceso de fabricación. (Pressman, 2005)

El control de la calidad tiene como primordial objetivo detectar y corregir los errores que van saliendo en todo el proceso de desarrollo del producto, además de asegurar que el producto final cumpla con los intereses del cliente.

1.4 Pruebas.

La prueba de software es un elemento crítico e imprescindible para la garantía de la calidad, y de ahí la necesidad de aplicarla. Se llama prueba de software al proceso en el que se ejecuta un sistema con el objetivo de detectar fallos. Es un proceso destructivo que determina el diseño de los casos de prueba y la asignación de responsabilidades. (Pressman, 2005)

El proceso de prueba conlleva a la realización de un conjunto de pruebas a lo largo del ciclo de vida del sistema. La Metodología del Proceso Unificado del Rational (RUP) propone una serie de pruebas durante todo el proceso de desarrollo del software, estas son (Jacobson, 2000):

- Prueba modular, prueba unitaria o prueba de componentes: Consiste en la prueba de cada módulo aislado del sistema.
- Prueba de integración: Se realiza a medida que los diferentes módulos del sistema se integran en uno, con el objetivo de comprobar que las interfaces entre los distintos módulos son correctas.
- Prueba del sistema: Se realiza cuando se han integrado todos los módulos, y su objetivo es comprobar que el sistema satisface los requisitos del usuario, tanto los funcionales como los no funcionales.

Las técnicas de prueba del software tienen el objetivo de diseñar pruebas que descubren diferentes tipos de errores con menos tiempo y esfuerzo. Estas técnicas se clasifican en:

- Pruebas de Caja Blanca: La elección de los casos de prueba se va a basar en el conocimiento que se tenga acerca de la estructura del objeto (diseño detallado, diagramas de flujo de datos y de control, código fuente).
- Pruebas de Caja Negra: La elección de los casos de prueba no se va a basar en el conocimiento que se tenga acerca de la estructura del objeto, sino en el conocimiento acerca de la funcionalidad deseada.

Las autoras de este trabajo señalan que todo proceso de desarrollo de software se encuentra sujeto a un mal entendimiento de los requisitos, a una mala comunicación por parte de los usuarios y a equivocaciones por parte de sus desarrolladores por lo que se hace necesario controlar lo que se realiza desde que se empieza a elaborar un producto. Esto se logra utilizando las pruebas, las cuales sirven para combatir estos errores y lograr calidad en lo que se está haciendo por parte de un determinado equipo de trabajo.

1.5 Métricas de Software

Paul Goodman en su libro *Métricas del Software, las mejores prácticas para el éxito en la gestión (Software Metrics-Best Practices for Successful IT Management)* define las métricas de software como: “La aplicación continua de mediciones basadas en técnicas para el proceso de desarrollo del software y sus productos para suministrar información relevante a tiempo, así el desarrollador junto al empleo de estas técnicas mejorará el proceso y sus productos”. Las métricas de software proveen la información necesaria para la toma de decisiones técnicas. (Goodman, 2004)

Por su parte, el Dr. Andújar agrega que el propósito de las métricas de software es permitir evaluaciones a través del ciclo de vida del software para asegurar que se alcanzan los requisitos del producto y de la organización. Las métricas del software ayudan a (Andújar , 2004):

- Seguir objetivamente a los proyectos, teniendo en cuenta impactos en el cumplimiento de plazos, costos y calidad (satisfacción del cliente).
- Estimar y predecir tendencias en función de la organización, los procesos, la gestión de cambio, la gestión de la configuración y medidas de software.
- Evaluar la productividad.
- Tomar decisiones objetivas.
- Comunicarse a través de un lenguaje común.
- Identificar los puntos críticos del proyecto para garantizarlos.
- Gestionar los riesgos.
- Identificar las mejores prácticas por comparación objetiva de métricas.

- Identificar y eliminar lo antes posible las causas de los principales defectos.
- Mejorar el entendimiento del proceso.
- Mejorar el proceso de software mediante una estrategia a medio y largo plazo.

1.6 Metodología RUP.

El Proceso Unificado de Desarrollo de Software (RUP) es una de las metodologías más empleadas a nivel mundial. Es un proceso de ingeniería de software propuesto por Corporación de Software Rational (*Rational Software Corporation*) para la construcción del ciclo de ingeniería de software, posibilitando la productividad en equipo y la ejecución de mejores prácticas de software a través de plantillas y herramientas que lo guían en todas las actividades de desarrollo del software. Es un contexto de trabajo genérico que puede aplicarse en disímiles sistemas de software, en múltiples áreas de aplicación, variados tipos de organizaciones y diferentes tamaños de proyecto.

La metodología RUP tiene como objetivo la producción de software de calidad en el plazo y a costos predecibles, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

1.6.1 Ciclo de Vida de RUP.

La metodología RUP tiene 3 características fundamentales:

- *Dirigido por Casos de Uso:* Los casos de uso reflejan lo que los usuarios futuros necesitan y desean, se captan cuando se modela el negocio y se representa a través de los requerimientos. A partir de aquí los casos de uso guían el proceso de desarrollo ya que los modelos que se obtienen como resultado de los diferentes flujos de trabajo, representan la realización de los casos de uso (cómo se llevan a cabo).
- *Centrado en la arquitectura:* La arquitectura muestra la visión común del sistema completo en la que el equipo de proyecto y los usuarios deben estar de acuerdo, por lo que describen los elementos del modelo que son más importantes para su construcción, los cimientos del sistema que son necesarios como base para comprenderlo, desarrollarlo y producirlo económicamente.

- *Iterativo e Incremental*: RUP propone que cada fase se desarrolle en iteraciones. Una iteración involucra actividades de todos los flujos de trabajo, aunque se desarrolla fundamentalmente algunos más que otros. Es práctico dividir el trabajo en partes más pequeñas o miniproyectos. Cada miniproyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en los flujos de trabajo y los incrementos al crecimiento del producto. Cada iteración se realiza de forma planificada, es por eso que se suele utilizar el término miniproyecto.

RUP divide el proceso en cuatro fases, dentro de las que se ejecutan diversas iteraciones en dependencia del proyecto (*Ver anexo 2*). Se representa el proceso en el que se grafican los flujos de trabajo y las fases mostrando la dinámica expresada en iteraciones y puntos de control (Jacobson, 2000).

1.6.2 Fases e Hitos de RUP.

1. *Concepción o Inicio*: Se describe el negocio y se delimita el proyecto, describiendo su alcance con la identificación de los casos de uso del sistema. Se debe mostrar una arquitectura candidata, estimar el coste en recursos de todo el proyecto, estimar los riesgos y las fuentes de incertidumbre. Esta fase culmina con un hito, determinar la visión del proyecto y definir claramente sus objetivos.
2. *Elaboración*: Su propósito es analizar el dominio del problema, definir, validar y cimentar la arquitectura, completar la visión, desarrollar el Plan de Desarrollo del Software y eliminar los mayores riesgos. En ella se obtiene un prototipo de la arquitectura, el que debe evolucionar hasta convertirse en el producto final. El hito de esta fase consiste en definir la arquitectura óptima del sistema.
3. *Construcción*: La finalidad de esta fase es alcanzar capacidad operacional del producto de forma incremental, a través de las sucesivas iteraciones. Se obtiene un producto listo para su utilización que está documentado y tiene un manual de usuario. El hito es tener la capacidad operacional del sistema.

4. *Transición*: El producto ya está listo para su instalación en las condiciones reales. Puede implicar reparación de errores. El hito de esta última fase es obtener el release del sistema. (Ver anexo 3). (Jacobson, 2000)

1.6.3 Definición de Roles según la metodología RUP.

La metodología RUP define roles de trabajo con sus correspondientes responsabilidades y la elaboración de planes para asegurar la calidad en el desarrollo del software. A continuación se especifican algunos de estos roles que ayudan a desarrollar las actividades de calidad de un proyecto. (RUP, 2003)

- Diseñador de Pruebas: es el encargado de modificar el plan de prueba y el script de prueba; define la propuesta de prueba, la configuración del ambiente de prueba y los elementos de prueba; es el responsable de la especificación de interfaz de prueba, de la configuración de entorno y de la suite de prueba; identifica los mecanismos de prueba y estructura la aplicación de la prueba.
- Revisor Técnico: lleva un registro de todas las revisiones, revisa el modelo de casos de uso del negocio, el modelo de análisis del negocio, los requisitos, la arquitectura, el diseño y el código.
- Probador: es el encargado de implementar las pruebas y la suite de prueba; ejecuta la suite de prueba, analiza fallos de prueba; es el responsable de los script de prueba y modifica la suite de prueba y la demanda de cambio.

Conjuntamente con los roles que propone RUP las autoras de este trabajo entienden necesaria la presencia de un administrador de la calidad quien será el encargado de dirigir y planificar las actividades orientadas al aseguramiento de la calidad.

1.7 Estándares de Calidad.

Los Estándares de Calidad son aquellos que permiten definir un conjunto de criterios de desarrollo que guían la forma en que se aplica la ingeniería del software. Suministran los medios para que todos los procesos se realicen de la misma forma y son una guía para lograr la productividad y la calidad.

La ventaja de los Estándares de Calidad es que la calidad se convierte en algo concreto, que se puede definir, que se puede medir y, sobre todo, que se puede planificar. Ayudan a comprender las relaciones que existen entre las diferentes características de un producto de software. Una desventaja es que no ha sido demostrada la validez absoluta de ninguno de estos estándares. Las conexiones que se establecen entre características, atributos y métricas se derivan de la experiencia, esto origina que existan múltiples Estándares de Calidad. En este epígrafe se hace referencia a algunos de ellos. (Scalone, 2006)

1.7.1 CMMI (Modelo de Madurez de Capacidad Integrado).

CMMI proporciona una fuente de información del nivel de desarrollo de software de las empresas en la práctica de los múltiples procesos, facilitando el inicio de la evaluación de la madurez y proporcionando una guía para desarrollar la estrategia para la mejora continua.

CMMI es el resultado de la publicación del modelo CMM con dos representaciones: continua y escalonada, ambas son equivalentes. La visión continua de una organización mostrará la representación de nivel de capacidad de cada una de las áreas del proceso del modelo. La visión escalonada definirá a la organización dándole en su conjunto un nivel de madurez del 1 al 5 (*Ver anexo 4*).

CMMI tiene 22 áreas de procesos que ayudan a evaluar el desarrollo de software de ingeniería de sistemas (CMMI-SE/SW). (CMMI, 2006)

✓ Aseguramiento de la Calidad según CMMI.

Dentro de las áreas de proceso que posee el Modelo Integrado de Capacidad y Madurez (CMMI) se encuentra el área de Aseguramiento de la Calidad de los procesos y los productos de trabajo. El modelo tiene como objetivo en esta área facilitar personas y gestión para lograr que los elementos y procesos de trabajo desempeñen los procesos establecidos.

Para alcanzar un producto confiable hay que tener en cuenta (CMMI, 2006):

- Evaluar objetivamente la ejecución de los procesos, los elementos de trabajo y servicios contra las descripciones de procesos, estándares y procedimientos.
- Identificar y documentar los elementos no conformes.
- Proporcionar información a las personas que están usando los procesos y a los gestores de los resultados de las actividades de aseguramiento de la calidad.
- Asegurar que los elementos no conformes sean arreglados.

Para asegurar la calidad CMMI requiere dos objetivos específicos que a su vez contienen dos prácticas específicas (CMMI, 2006):

- Objetivo Específico: Evaluar objetivamente los procesos y los productos del trabajo.
 - Evaluar objetivamente los procesos.
 - Evaluar objetivamente los productos y los servicios del trabajo.
- Objetivo Específico: Proporcionar un entendimiento objetivo.
 - Comunicar y asegurar la resolución de las no conformidades emitidas.
 - Establecer los registros.

El aseguramiento de la calidad es un área de proceso clave, que en ocasiones no se le da la importancia suficiente ni se le dedica el tiempo necesario pero sin ella es casi imposible crear un modelo de calidad.

1.7.2 Estándar ISO 9000-3.

La Organización Internacional para la Estandarización, mejor conocida como ISO, es la agencia especializada en estandarización, conformada por representantes de los cuerpos normalizadores.

La ISO 9001 fue escrita para ser utilizada por toda clase de industrias, es regularmente difícil interpretarla para el desarrollo de software, por lo cual se publicó la ISO 9000-3 "Guía para la aplicación de ISO 9001 para el desarrollo, implementación y mantenimiento de software".

El objetivo de la ISO 9000-3 es proveer las especificaciones de cómo aplicar la ISO 9001 al desarrollo del software, implementación y mantenimiento. Se incluyen algunos temas que no se encuentran en las normas ISO 9000 genéricas, tales como Administración de la Configuración o Planeación de Proyectos. Sería poco probable lograr resultados de calidad en un proyecto de desarrollo de software de tamaño mediano, sin haber tomado las provisiones necesarias para el control de configuración. Esto implica que para ciertos productos o servicios, la especificación de requerimientos contenida en las normas genéricas ISO 9000 no es suficiente para asegurar la calidad, y esto justifica la necesidad de otras normas o guías más específicas. (Navarro, 2003)

Dentro de los beneficios que se obtienen de la certificación ISO 9000-3, se encuentran:

- Mejor documentación de los sistemas.
- Cambio cultural positivo.
- Incremento en la eficiencia y productividad.
- Mayor percepción de calidad.
- Se amplía la satisfacción del cliente.
- Se reducen las auditorías de calidad de los clientes.
- Agiliza el tiempo de desarrollo de un sistema.

✓ **Aseguramiento de la Calidad según el modelo ISO 9000-3.**

Para el estándar ISO 9000-3 el aseguramiento de la calidad tiene como finalidad proporcionar la seguridad de que los productos y los procesos en el ciclo de vida del proyecto están de acuerdo a los requisitos específicos y que se ajustan a los planes establecidos. El aseguramiento de la calidad puede ser interno o externo dependiendo de si la evidencia de la calidad del producto o del proceso se demuestra a la gerencia del proveedor o del cliente. Este proceso implementa varias actividades como (Bañeres):

- Implementación del proceso: Esta actividad cuenta con varias tareas como: establecer un proceso de aseguramiento de la calidad ajustado al proyecto. Elaborar, documentar, implementar y mantener actualizado un plan de ejecución

de las actividades y tareas del proceso. Implementar normas, metodologías, procedimientos y herramientas para asegurar la calidad. Proporcionar al cliente los registros de las actividades y tareas de aseguramiento de la calidad.

- Aseguramiento del producto: Dentro de esta actividad se encuentran diversas tareas como: asegurar que todos los planes requeridos por el contrato estén documentados. Asegurar que todos los productos de software usados y la documentación relacionada con ellos cumplan lo establecido. En la preparación de la entrega de los productos, deberá ser asegurado que ellos satisfacen totalmente sus requisitos contractuales y que son aceptables para el cliente.
- Aseguramiento del proceso: Esta actividad está formada por las siguientes tareas: asegurar que los procesos del ciclo de vida del producto cumplen con lo establecido y se ajustan a los planes. Asegurar que las prácticas internas del diseño del producto, del ambiente de desarrollo y del ambiente de prueba cumplen lo establecido en el contrato.
- Aseguramiento del sistema de calidad. Esta actividad la integran un grupo de tareas que se mencionan a continuación: asegurar actividades adicionales de gestión de la calidad de acuerdo con las cláusulas de la norma ISO 9001.

CAPÍTULO 2: DESCRIPCIÓN Y SITUACIÓN ACTUAL DEL PROYECTO SIMULADOR QUIRÚRGICO.

Introducción

Para la realización de una estrategia de aseguramiento de la calidad es necesario efectuar un análisis de la situación actual en la que se encuentra el proyecto en cuanto al área de Aseguramiento de la Calidad, por lo que este capítulo estará dedicado a estudiar el estado real del proyecto productivo Simulador Quirúrgico en este sentido.

Primeramente se realizará una descripción del proyecto donde se aborda su necesidad, objetivos, metas y propósito, la que proporcionará información sobre qué es lo que se pretende con su desarrollo. Seguidamente se especifican las herramientas que son utilizadas para el desarrollo del Simulador Quirúrgico y las condiciones que debe cumplir, además de la realización de una descripción minuciosa del proceso de aseguramiento de la calidad en el proyecto.

2.1 Descripción del Simulador Quirúrgico.

La Cirugía de Mínimo Acceso (CMA) ha sustituido a la cirugía abierta en un número considerable de procedimientos, este auge se debe a las grandes ventajas que tiene tanto para el médico como para el paciente. En ella el cirujano visualiza a través de un monitor, una “información electrónica equivalente” a la imagen real de la anatomía del paciente, mientras efectúa la intervención con delgados y versátiles instrumentos que son introducidos en las cavidades corporales mediante pequeñas incisiones o del propio endoscopio. Esto atribuye tensiones específicas al cirujano por lo que necesita una coordinación psicomotora y de habilidades con la nueva interfaz video-endoscópica.

Todo lo anterior permite comprender la necesidad de que exista una solución para este problema por lo que surge el proyecto Simulador Quirúrgico, que contribuirá a la apertura del camino de la realidad virtual aplicada a la medicina, obteniéndose un simulador para el entrenamiento y la planificación de intervenciones íntegramente cubano, que será insertado de manera gradual en el proceso de enseñanza de los procedimientos endoscópicos en nuestro país.

Capítulo II: Descripción y Situación Actual

El *propósito* que se persigue con la obtención del Simulador Quirúrgico es obtener un prototipo cubano que aplique la Realidad Virtual al proceso de enseñanza de Cirugía de Mínimo Acceso y Endoscopia. Esta tecnología debe permitir exigencias visuales y ejercicios de entrenamiento para el desarrollo de habilidades, los cuales tendrán diferentes niveles de complejidad que irán aumentando el rigor y nivel de realismo a medida que el usuario avance en la ejecución hasta el punto de tener la posibilidad de planificar y repasar virtualmente una operación con modelos 3D generados a partir de información real del paciente obtenido por equipos TAC o MRI. El producto también tendrá un módulo de registro en el cual los usuarios gestionan sus datos y sus evaluaciones en el entrenamiento.

El tradicional método de aprendizaje de la cirugía, conocido como método de compañerismo, donde el cirujano en formación observa al profesor, luego realiza sus operaciones reproduciendo los gestos bajo supervisión, no enfatiza en la adquisición de las habilidades necesarias para la práctica de la CMA y por otra parte, los errores que cometen los adiestrados sobre los pacientes durante su aprendizaje.

Debido a toda esta problemática nace la *necesidad* de buscar soluciones eficientes que estén a tenor con el desarrollo tecnológico actual, y que permitan revolucionar metodológicamente y científicamente el proceso de enseñanza en la educación quirúrgica, mediante la aplicación de técnicas modernas de aprendizaje, tales como la simulación virtual, que se vislumbra como la tecnología emergente aplicada a la enseñanza quirúrgica de mayor importancia.

El *objetivo general* es crear 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 de la endoscopia diagnóstica.

Este proyecto está vinculado con las investigaciones que se realizan en el país como parte del programa que se lleva a cabo en el sector de la salud.

Se desea que el producto sea utilizado de manera independiente por todas las instituciones cubanas de salud, con servicios de Cirugía de Mínimo Acceso y de Endoscopia.

2.2 Herramientas y condiciones de desarrollo en el proyecto productivo Simulador Quirúrgico.

Para un funcionamiento óptimo del Simulador Quirúrgico el grupo de desarrollo en coordinación con el cliente decidió que debe tener una alta velocidad de procesamiento, tiempo de respuesta, de recuperación y disponibilidad. Además la cantidad de Cuadros Por Segundos (FPS) no debe ser inferior a 35.

Inicialmente se realizaba solo la versión para Windows debido a que el producto se implantará en todos los hospitales del país y se sabe que aún no se ha logrado un buen trabajo en Linux, a pesar de esto posteriormente se decide realizar una versión en Linux, producto del proceso de migración hacia el software libre y de la importancia y necesidad de producir una versión libre.

El hardware debe ser compatible con tarjetas gráficas de alta capacidad de procesamiento, por lo que actualmente se está utilizando tarjetas gráficas de la familia NVIDIA (Quadro FX 500/FX 600), además de poseer un procesador superior a los 2 GHz de velocidad y una RAM superior a los 512MB. Se usará transparentemente el engine gráfico STK en su versión 2.3 como framework en el desarrollo del Simulador Quirúrgico.

Todo el código será en lenguaje C/C++, regido por la teoría de Programación Orientada a Objetos. En Windows se utilizará para la implementación el Visual Studio 2003 y como IDE de Trabajo, el Visual Paradigm para el modelado, el 3DMAX en el diseño y el SQL Server como gestor de Base de Datos. En el Sistema Operativo Linux se utilizará el Code Blok para desarrollar, el Visual Paradigm como herramienta de diseño de clases y Postgresql como gestor de Base de Datos.

Como sistema de control de versiones se utiliza el Subversion, que es libre y de código fuente abierto permitiendo la recuperación de versiones antiguas de los datos del proyecto o examinar el historial de los cambios. Cada integrante del proyecto tiene un usuario en el Subversion que le permite descargar los archivos de su interés y al finalizar su trabajo salva la información modificada, los ficheros deben tener un tamaño aproximado de 100 MB.

También se utilizará el Trac como una solución para garantizar la administración de la configuración de software pues constituye un sistema de Wiki mejorado que posee una interfaz para el Subversion mediante la cual se puede examinar el código fuente que está bajo este.

2.3 Proceso de Desarrollo en el Simulador Quirúrgico.

El proceso de desarrollo del Simulador Quirúrgico se encuentra dividido en 4 fases o etapas de desarrollo. En cada una de estas fases se obtiene una versión funcional a partir de la cual se podrán entrenar las diferentes habilidades a desarrollar en los médicos cirujanos, desde el entrenamiento de habilidades básicas hasta el entrenamiento en un simulador profesional en la cuarta y última etapa. Se obtendrá además la documentación y ayuda necesaria para el entendimiento y posible uso de cada entregable (Ver Figura 2.1).

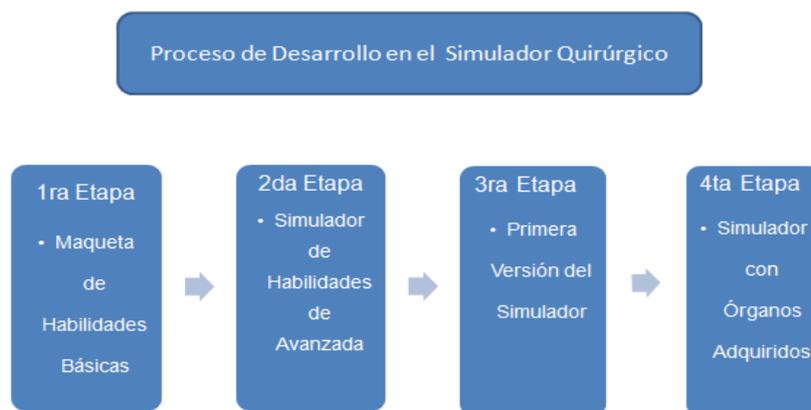


Figura 2.1: Estructura de desarrollo del proyecto Simulador Quirúrgico.

A continuación se detalla cada una de las fases o etapas de desarrollo del proyecto:

- **Primera Etapa**

Es una maqueta de habilidades básicas la que tiene como objetivo posibilitar la realización de ejercicios básicos sobre un ambiente no anatómico desarrollando habilidades como el movimiento de objetos, la manipulación de cámaras y la coordinación mano-ojo.

- **Segunda Etapa**

Simulador de habilidades de avanzada a partir de la imitación de procesos, incluyendo endoscopía, tiene como objetivo desarrollar habilidades básicas de mayor complejidad como son corte, sutura y grapado.

- **Tercera Etapa**

Primera versión del simulador a partir de la generación de órganos virtuales teniendo como principal objetivo la obtención de un simulador para el desarrollo de habilidades básicas con ambiente más realista.

- **Cuarta Etapa**

Simulador con órganos adquiridos a partir de scanner (TAC, MRI) y retroalimentación al tacto el que tiene como objetivo obtener un simulador que permita el desarrollo de habilidades básicas y avanzadas con efectos gráficos de alta complejidad.

En cada una de las etapas del Simulador Quirúrgico se precisaron las fases del proceso de desarrollo basándose en la metodología RUP. En cada fase se planificó un número de iteraciones en dependencia del rigor que posea, en el caso de la primera etapa del proyecto en la fase de inicio se realizará 1 sola iteración ya que con eso es suficiente para obtener una visión del sistema y especificar los requerimientos. Habrá 2 iteraciones en la fase de elaboración para refinar bien el diseño de cada artefacto y asegurar en buena medida la calidad del producto. En la fase de construcción se realizarán 3 iteraciones con el objetivo de implementar los ejercicios diseñados en la fase de elaboración correspondientes a la primera versión del Simulador. Luego se entrará en la fase de transición en la que se realizará 1 iteración, para establecer la retroalimentación por parte del cliente y a partir de esto se detectan las no conformidades del producto que conformarán posibles mejoras para elevar su aceptación.

2.4 Descripción del proceso de Aseguramiento de la Calidad en el Simulador Quirúrgico.

Actualmente como parte de las acciones del proceso de Aseguramiento de la Calidad en el proyecto productivo Simulador Quirúrgico se tiene definido un plan de Aseguramiento de la Calidad, el cual se encuentra formado por un conjunto de actividades como se muestra a continuación (Ver Figura 2.2):

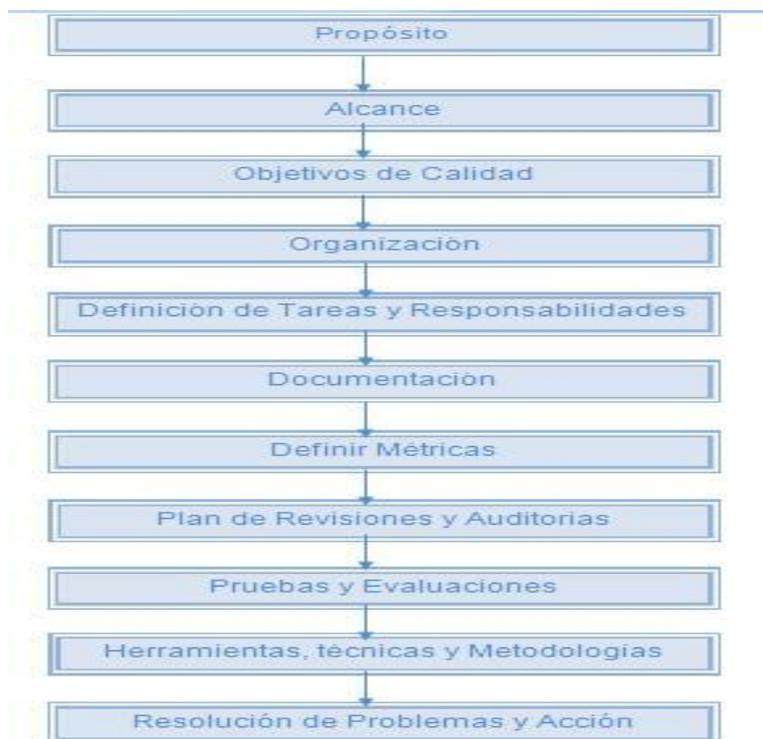


Figura 2.2: Actividades del Plan de Aseguramiento de la calidad del Software en el Simulador Quirúrgico.

Propósito: Cuando se habla del propósito del plan de Aseguramiento de la calidad se están definiendo los pasos a seguir para alcanzar un nivel elevado de calidad en el software desarrollado y en los procesos implicados.

Actualmente el propósito que se persigue con el plan de aseguramiento de la calidad es especificar en un solo documento como se asegurará la calidad del proyecto “Simulador Quirúrgico”, con el fin de proporcionar la adecuada seguridad de que el

Capítulo II: Descripción y Situación Actual

producto a obtener y los procesos del ciclo de vida del proyecto estén conformes con los requisitos específicos y se ajusten a los planes establecidos.

Alcance: En el alcance se precisan los proyectos con el que se involucra el plan.

El Plan de Aseguramiento de la calidad del Simulador Quirúrgico abarca solamente las revisiones, pruebas y auditorías que se realizan.

Objetivos de Calidad: En los objetivos de calidad se tienen en cuenta los requisitos del proyecto conjuntamente con los requisitos de calidad. A continuación se especifican algunos de estos objetivos de calidad definidos en el Simulador Quirúrgico.

- Asegurar la calidad desde el inicio del proyecto para cumplir con los requerimientos del cliente, tomando como referencia los procedimientos de la UCI, normas internacionales existentes y el área de proceso del estándar de calidad CMMI: Aseguramiento de la Calidad del Producto y el Proceso.
- Cumplir con las exigencias del modelo CMMI (Nivel 2-3).
- Aplicar las mejores prácticas de ingeniería y gestión de software.
- Lograr que todo el personal cuente con los conocimientos y habilidades necesarias para realizar las tareas y actividades solicitadas por el cliente.

Al concebirse un proyecto se debe iniciar el proceso de aseguramiento de la calidad donde se organiza la estructura del equipo de calidad y se define cada uno de los roles correspondientes, seguidamente se imparte un curso de capacitación y nivelación para los miembros del equipo; en el Simulador Quirúrgico el proceso de aseguramiento de la calidad no se inició de esta forma, no se define el equipo de calidad al inicio del proyecto, y luego cuando se forma dicho equipo no se le impartieron los cursos correspondientes y necesarios para que desempeñen su función con los conocimientos básicos que debe tener cada rol, trayendo como resultado que al proceso de aseguramiento de la calidad se le encuentren deficiencias que influyan en la calidad del producto final. Otros de los objetivos al que no se le da cumplimiento es el de cumplir las exigencias del modelo CMMI ya que el proyecto no se rige por ningún modelo de calidad a la hora de asegurar la calidad de los procesos y productos.

Organización: Muestra la organización del equipo de calidad del proyecto Simulador Quirúrgico con el objetivo de mostrar los involucrados en el desarrollo del mismo.

El equipo de calidad del proyecto es el encargado de realizar una estrategia de aseguramiento de la calidad interna, teniendo en cuenta las necesidades de aseguramiento de la calidad. Los roles de los que se disponen actualmente en el Simulador Quirúrgicos son: Responsable de la Calidad, Diseñador de Casos de Pruebas, Auditor de la calidad y el probador. (Ver Figura 2.3)



Figura 2.3: Estructura del Equipo de Calidad del Simulador Quirúrgico.

Esta estructura definida para el equipo de calidad del proyecto productivo Simulador Quirúrgico no es la más conveniente para su desarrollo, se define un rol de auditor que no desempeña función alguna debido a que las auditorías que se mencionan en el plan de aseguramiento de la calidad son planificadas y realizadas por la Dirección de calidad de software de la Universidad. Sin embargo se cree necesaria la presencia del rol de revisor técnico formal el cual es el encargado de la realización de las revisiones y de registrar sus resultados luego de efectuarse. Por último, otra dificultad a señalar en la organización del equipo de calidad es que no se especifican los conocimientos básicos que debe disponer el individuo para desempeñar un rol en específico y de esta forma poder cumplir con las responsabilidades correspondientes.

Capítulo II: Descripción y Situación Actual

Tareas y Responsabilidades: En este punto se especifican las tareas y responsabilidades del equipo de aseguramiento de la calidad dentro del proyecto que se realizarán en el transcurso de vida.

En el Proyecto Simulador Quirúrgico se definieron un grupo de actividades como son:

- Auditoría de la arquitectura:
Esta tarea tiene como objetivo revisar la arquitectura del sistema tomando como referencia las listas de chequeo elaboradas a partir de los lineamientos de calidad y la norma IEEE 1471, será realizada en el transcurso de la etapa de análisis y diseño de cada iteración y la misma la realizará el planificador o el responsable de la calidad del proyecto.
- Revisión de diseño:
Tiene como objetivo revisar el diseño del sistema tomando como referencia las listas de chequeo elaboradas a partir de los lineamientos de calidad, la norma IEEE 1016 y la ISO 12207, será realizada en el transcurso de la etapa de análisis y diseño de cada iteración.
- Pruebas de caja blanca:
A partir de esta actividad se realiza una comprobación de la calidad del código a partir del diseño de caso de pruebas y su puesta en práctica, tomando como referencia la ISO 9126, ISO 12207, ISO 12119, IEEE 1028. Esta tarea la realizarán el planificador o el responsable de la calidad durante la implementación.
- Prueba de caja negra:
En esta tarea se realiza una comprobación de los requisitos de calidad a través del diseño de casos de prueba y su puesta en práctica, tomando como referencia la norma ISO 9126 y la IEEE 1028, se realizará al final de cada iteración por el planificador o el responsable de la calidad.
- Auditoría a la Gestión de configuración:
En esta actividad se verificará que se estén controlando los cambios, y que antes de cada cambio significativo se realice revisiones técnicas formales, además de comprobar que los cambios se están informando oportunamente y verificar la conformidad con los lineamientos de calidad y las normas ISO 12207,

Capítulo II: Descripción y Situación Actual

ISO 15486 e IEEE 828. Se realizará durante la implementación por el planificador o el responsable de la calidad.

- Revisión de la documentación, revisiones técnicas formales:

Esta actividad se realiza con el objetivo de que se revise toda la documentación generada, una buena práctica es enviando los documentos elaborados al resto de los miembros del proyecto hasta lograr el consenso grupal. Además, cada artefacto que se obtenga en el proyecto se le realizará una revisión técnica formal, teniendo en cuenta ISO 9127. Será realizada cada vez que se concluya la elaboración de cada documento y será realizada por el líder del proyecto, planificador o el responsable de la calidad.

- Revisión Post-Morten:

Será ejecutada al finalizar cada fase de proyecto, en la que se evaluará el desarrollo de las actividades y facilitar las recomendaciones para próximas fases. Será realizada por el líder del proyecto al finalizar cada fase del proyecto.

Estas tareas se planificaron y no fueron ejecutadas por existir déficit de personal capacitado para su realización ya que el personal con el que cuenta el proyecto no tiene la preparación ni el conocimiento suficiente para su ejecución, además de no contar con un plan de capacitación para poder adquirir dominio en el tema.

Se planificaron varias auditorías que no son responsabilidad del equipo de calidad sino de la Dirección de Calidad de Software de la UCI, entre ellas se encuentran: auditoría de la arquitectura y auditoría de la gestión de configuración.

Todas las actividades no pueden ser realizadas por el responsable de la calidad, para eso se encuentra el equipo de aseguramiento de la calidad donde cada rol es responsable de realizar un conjunto de tareas que se le asignan, el responsable de la calidad solo guiará su realización, todo esto es producido por una mala organización del equipo y asignación de las tareas, sobrecargando de trabajo al responsable de la calidad del proyecto.

Se habla sobre tomar como referencia las listas de chequeo, pero dichas listas no se han elaborado, por lo que no se pueden utilizar para guiarse en la realización de las revisiones.

Capítulo II: Descripción y Situación Actual

Documentación: En este punto se enumeran los documentos que guían el desarrollo del proyecto y que conviene ser revisados como parte de las actividades de aseguramiento de la calidad.

La documentación existente no abarca todas las plantillas necesarias, ejemplo de esto es que se hace referencia a una lista de no conformidades de cada una de las tareas planificadas, entre las que se encuentran revisiones, auditorías y pruebas pero como dichas tareas no se realizan no existe ninguno de estos documentos, además no se hace mención del Plan de Desarrollo que es uno de los documentos más importantes en la realización del Plan de Aseguramiento de la Calidad.

Definir Métricas: En esta actividad se describe para cada propiedad de calidad las métricas que se utilizarán para medir dicha propiedad como producto del monitoreo del trabajo.

En el Simulador Quirúrgico no se realizan actividades de estimación y medición, aunque eso fuera lo ideal ya que las métricas permiten que se mejore el proceso de software, así como la planificación y evaluación de la calidad del producto.

Plan de Revisiones y Auditorías: Se explica brevemente cada tipo de revisión y auditoría que se realiza en el proyecto. Para cada tipo se identifica los artefactos del proyecto que serán el asunto de la revisión o auditoría.

En el Plan de Aseguramiento de la Calidad del proyecto se hace mención a un Plan de Revisiones y Auditorías que no está definido. No aparecen una serie de puntos importantes entre los que se encuentran: el cronograma donde se planifica cada una de las revisiones y auditorías, la organización del equipo que será el responsable en cada una de las actividades programadas, la resolución de problemas y actividades correctivas, las herramientas, técnicas y metodologías, a pesar de que en otros puntos dicen utilizar las listas de chequeo como herramienta y como técnica las revisiones técnicas formales.

En general el Proceso de Aseguramiento de la calidad en el Simulador Quirúrgico se encuentra en un estado poco favorable, ya que contiene una serie de incongruencias que pueden tener como consecuencia que el producto final no posea la calidad requerida. Desde su inicio no se conforma un equipo de calidad, dando como resultado

Capítulo II: Descripción y Situación Actual

que en el levantamiento de requisitos no se tenga en cuenta las normas de calidad requeridas ni sea realizado contando con la presencia de alguno de los miembros del equipo. El proyecto no se rige por un modelo o estándar de calidad pero sí tienen en cuenta los Lineamientos Mínimos de Calidad. No existe una buena relación entre el cliente y los miembros del proyecto. Se han realizado algunas pruebas de caja blanca aunque no se tiene ningún registro de ello, lo que demuestra que en el proyecto no se lleva a cabo el proceso de registro de las no conformidades encontradas.

En el Simulador Quirúrgico se encuentra definido un Plan de Aseguramiento de la Calidad pero este no da solución a todas las problemáticas que pueden surgir. En él se mencionan una serie de tareas que no se llevaron a cabo en esta primera etapa de vida, además de faltarle algunas secciones como son el plan de revisiones, su cronograma con las fechas principales del proyecto así como revisiones que son provocadas por la entrega de artefactos, la organización del equipo que realizará las revisiones, no se hace mención a la resolución de problemas y las actividades de corrección que no es más que el procedimiento utilizado para informar y manejar los problemas que fueron identificados en el proceso de revisión del proyecto. Tampoco se definió las herramientas, técnicas y metodologías específicas que serán utilizadas para llevar a cabo las actividades de revisión del plan, además se debe describir las listas de chequeo a utilizar en cada revisión. Otro de los puntos que no se hace referencia es el de las herramientas, técnicas y metodologías.

Cuando se habla del Proceso de Aseguramiento de la Calidad no se puede centrar únicamente en el Plan de Aseguramiento de la Calidad, pues el plan solamente es una de las actividades que se realizan. Hay que tener en cuenta cómo se efectúa la gestión de configuración del software en el proyecto donde actualmente se tiene el Subversion (SVN) pero se está utilizando solo para el código, también se debe tener presente la gestión para controlar las versiones de los demás artefactos del proyecto como por ejemplo el expediente y otros documentos, para esto se está configurando el TRAC.

2.5 Resultados de Auditoría Realizada al proyecto Simulador Quirúrgico.

En una auditoría efectuada al proyecto por la Dirección de Calidad del Software de la UCI se detectaron una serie de no conformidades que pudieran afectar el desarrollo del proyecto, las cuales se mencionan a continuación: en la gestión de requisitos no se

Capítulo II: Descripción y Situación Actual

incluyó la descripción de los conceptos tratados en el modelo de dominio y no se lleva el control y seguimiento de los cambios en los requerimientos; en cuanto al registro del capital humano y los recursos materiales se detectó que algunas de las PC del proyecto no cuentan con el hardware necesario para desarrollar este tipo de proyecto; en la definición de la arquitectura existe una falta de detalle en los objetivos y restricciones arquitectónicas, el tamaño y el rendimiento de la aplicación, casos de usos arquitectónicamente significativos, los elementos del diseño, la vista de procesos y los indicadores de calidad; respecto a la definición y empleo de las herramientas a utilizar en el desarrollo del proyecto la estructura del servidor SVN no se complementa con la estructura del expediente de proyecto; en el registro de las estimaciones realizadas no se usan métodos para estimar esfuerzos; en cuanto al registro de Actividades Conjuntas y visitas de los diferentes factores del proyecto no se realizan reuniones con el cliente por razones ajenas a los miembros del equipo de proyecto; respecto a la vinculación con los procesos docentes e investigativos existe una falta de relación con el expediente productivo orientado por el Departamento Docente Central de Práctica Profesional (DDC de PP); en la definición y empleo de las herramientas a utilizar en el desarrollo del proyecto no se utiliza ninguna para la gestión documental y ni para el control y seguimiento de errores; en el plan de aseguramiento de la calidad hacen referencia al documento Plan de mediciones que no está incluido en el expediente de proyecto; respecto al establecimiento de la estructura organizacional, los roles y las responsabilidades se describen roles que no están presentes en el proyecto y se asignan otros no descritos y por último en el glosario de términos del proyecto no se encuentran todos los términos necesarios como STK.

Luego de realizarse la auditoría por parte de la Dirección de Calidad del Software de la UCI el proyecto aplicó algunas acciones correctivas para resolver las no conformidades detectadas entre estas se encuentran: se cambió la estructura del servidor Subversion de manera tal que se complementa con la estructura del expediente del proyecto y se comenzó a utilizar el TRAC como herramienta para la gestión documental del proyecto.

CAPÍTULO 3: PROPUESTA Y DESCRIPCIÓN DE LA SOLUCIÓN.

Introducción.

Este capítulo estará dedicado a describir y explicar la estrategia de aseguramiento de la calidad de software para el proyecto productivo Simulador Quirúrgico que será aplicada a partir de la segunda etapa de desarrollo de dicho proyecto partiendo de un análisis previo realizado al proceso de aseguramiento de la calidad en la primera etapa efectuado en el capítulo anterior.

Dentro de esta estrategia se abordan aspectos como son la organización de un equipo de calidad, la elaboración de un plan de Aseguramiento de la calidad, el proceso de revisiones y pruebas dentro del proyecto Simulador Quirúrgico y por último se lleva a cabo una validación de la propuesta.

3.1 Características de la Estrategia

La estrategia de aseguramiento de la calidad del proyecto productivo Simulador Quirúrgico que se concreta en el presente capítulo comienza con una definición adecuada de los roles que conformarán el equipo de calidad del software; luego se confecciona el plan de aseguramiento de la calidad, artefacto que suministra una visualización de cómo debe ser seguro el producto y la calidad del proceso. Con la realización de este plan se pretende desplegar un modelo para planificar y sistematizar las acciones con el objetivo de proporcionar seguridad según los requerimientos establecidos por el proyecto. El Plan de Aseguramiento de la calidad se realiza con el objetivo de brindar un seguimiento a las actividades de ingeniería del software, describiendo y especificando cómo se asegurará la calidad del proyecto Simulador Quirúrgico. Dentro de estas actividades se encuentran las revisiones donde se especifica el momento en el que se realizarán, así como una breve explicación de cómo fluirá el proceso de revisiones durante el proceso de desarrollo. Otras actividades que forman parte de la estrategia son diseñar las listas de chequeo correspondientes a aquellas actividades dentro del proceso de aseguramiento de la calidad, registrar todas las no conformidades encontradas ya que a partir de estas se elabora un registro de Respuesta a las no conformidades para proporcionar seguimiento a las deficiencias encontradas, y por último pero no menos importante se confecciona un plan de pruebas en el que se planifica de manera anticipada las pruebas que se le aplicarán a cada uno de los artefactos que crea el producto durante su desarrollo realizando una descripción de cómo se efectuará dicho proceso de pruebas.

A raíz de la confección de la estrategia de aseguramiento de la calidad se crearán estos documentos utilizando las plantillas propuestas por la Dirección de Calidad de Software de la UCI.

3.2 Definición del Equipo de Aseguramiento de la Calidad en el Simulador Quirúrgico.

El equipo de aseguramiento de la calidad del Simulador Quirúrgico es el encargado de realizar la estrategia de aseguramiento de la calidad interna del proyecto, teniendo en cuenta las necesidades de aseguramiento de la calidad que se tiene. Dicho equipo se organizará por roles según la metodología RUP añadiéndole el rol de administrador de la calidad, además se definirán las actividades que van a desempeñar cada uno y la base de conocimientos de la que deben partir.

Uno de los roles que se propone es el **Administrador de la Calidad**, encargado de planificar el proceso de aseguramiento de la calidad en el proyecto y de realizar el plan de prueba de cada iteración. El administrador de calidad es el responsable de coordinar el proceso de recopilación, análisis y reporte de las estadísticas de calidad, realizar el seguimiento a los planes de Calidad y de Medición involucrando la recogida de los datos que sean necesarios, determinar el avance logrado, analizar las desviaciones de los planes y coordinar las acciones que se deriven de ese análisis. Este administrador de la calidad guiará una serie de actividades como son las revisiones técnicas formales, las pruebas que se le realicen al proyecto, manejará todo lo relacionado con los riesgos de calidad y realizará el resumen de la evaluación de las pruebas.

El rol **administrador de la calidad** lo debe ocupar una persona con conocimientos básicos de calidad e ingeniería de software, entendimiento del negocio que se está modelando en el proceso de software, además debe tener conocimientos sobre la metodología de desarrollo RUP y el lenguaje de modelado UML (*Unified Modeling Language*), debe conocer sobre las métricas y el análisis estadístico, el monitoreo de proyectos informáticos y los estándares CMMI o ISO 9000.

Otro rol que se propone es el **Diseñador de casos de pruebas**, encargado de diseñar los casos de pruebas, identificar las técnicas apropiadas, las herramientas e instrucciones para la implementación de las pruebas y de definir las listas de chequeo que se deben aplicar. Sus conocimientos básicos están encaminados hacia la metodología RUP y el modelado UML, además de los conocimientos en calidad del software, estándares y modelos, estrategias de pruebas y pruebas de software.

También se propone el rol de **Probador** encargado de ejecutar las pruebas diseñadas por el diseñador de casos de pruebas y de registrar los resultados obtenidos. Se precisa que tenga conocimiento del negocio, de calidad del software, estándares y modelos y las pruebas de software.

De la misma forma que una persona se especializa en el tema de las pruebas existe un rol que se encarga de las revisiones técnicas, por lo que se propone para el proyecto un **Revisor Técnico Formal**, encargado de la elaboración de las listas de chequeo para las revisiones, de revisar que los artefactos que se generan en el proyecto concuerden con lo establecido y de registrar los resultados de las revisiones. Sus conocimientos básicos están enmarcados fundamentalmente en la ingeniería del software, calidad del software, estándares y modelos y en las revisiones técnicas formales.

Una vez que se han precisado los roles que intervendrán en la estrategia de aseguramiento de la calidad del proyecto es necesario definir un plan de entrenamiento para lograr nivelar los conocimientos básicos para realizar las actividades de aseguramiento de la calidad. La planificación del entrenamiento debe ser desarrollada por el Administrador de la Calidad teniendo en cuenta los conocimientos básicos requeridos por cada rol para que puedan desempeñar sus tareas de la mejor manera posible.

Se considera que entre los contenidos impartidos no deben faltar los temas relacionados con Evaluación de Software, Pruebas de Software y Métricas basados en normas y estándares internacionales, como propuesta inicial. A medida que se desarrolla el proyecto y en función de las necesidades, se podrán ir tramitando otros temas de capacitación relacionados con el negocio que se modela.

3.3 Definición del Plan de Aseguramiento de la Calidad del Software en el Simulador Quirúrgico.

Apoyándose en las plantillas propuestas por la Dirección de Calidad de la UCI, se implementa el plan de aseguramiento de la calidad que tiene en su contenido varias secciones, estas van a facilitar la organización y comprensión de todas las actividades de aseguramiento de la calidad en el proyecto.

En esta actividad es de vital importancia determinar los objetivos de calidad, los que deben estar en correspondencia con los objetivos del grupo de desarrollo, se definen

los estándares y guías que le dan validez al documento además de proponer una métrica encaminada a estudiar el comportamiento de los defectos en el proyecto. En el plan de Aseguramiento de la calidad se plasman las revisiones a realizar en el proyecto donde se señala el momento exacto en el que corresponde realizarle cada revisión precisando las herramientas que son utilizadas para su ejecución. También se mencionan las acciones correctivas y preventivas como parte de la mejora de la calidad, el plan de pruebas y el listado de los registros de calidad que serán utilizados en el transcurso de vida del proyecto.

El propósito que se busca con este Plan de Aseguramiento de Calidad es el de describir y especificar cómo se asegurará la calidad del proyecto Simulador Quirúrgico con el objetivo de proveerle seguridad al producto a obtener y que los procesos del ciclo de vida del proyecto estén acorde con los requisitos específicos y coincidan con los planes establecidos. El plan incluye las revisiones y pruebas que se efectuarán en los diferentes períodos del proceso de Desarrollo del Simulador Quirúrgico.

Para la confección de este plan se referencia una serie de documentos que se mencionan a continuación:

- Lineamientos Mínimos de Calidad.
- Plan de Gestión de Riesgos.
- Plan de Gestión de Configuración.
- Plan de Desarrollo de Software.
- Plan de Pruebas.

3.3.1 Objetivos de calidad para el proyecto.

Los objetivos de calidad que se proponen son:

- Preservar el trabajo guiándose por los diversos estándares y normas internacionales (CMMI), así como con la metodología establecida por RUP.
- Gestionar los riesgos a través de la lista de riesgos y del plan de mitigación de riesgos.
- Garantizar que el trabajo que se realice cumpla con los niveles de calidad definidos por medios tales como la comprobación y valoración sistemática en el desarrollo del proyecto.

- Asegurar que el equipo de calidad disponga de personal preparado para la ejecución de las tareas necesarias para alcanzar la calidad del proyecto.
- Velar por la aplicación de las mejores prácticas de Ingeniería y Gestión de Software.
- Estar en correspondencia con los Lineamientos Mínimos de Calidad.

3.3.2 Métrica para el proceso de revisiones de software en el Simulador Quirúrgico

En estudios realizados se ha demostrado que las revisiones de software son un potente método para la detección de defectos, encuentran de un 60 a un 90% de todos los defectos, y además proveen retroalimentación que permitirá a los desarrolladores de software evitar la introducción de defectos a trabajos futuros (Schulmeyer, 1997).

Los Líderes de Proyectos que recopilen y usen efectivamente la información de las revisiones pueden:

- Asignar recursos.
- Controlar, según los procedimientos establecidos por el proyecto.
- Determinar la calidad del software revisado.
- Medir la efectividad de las Revisiones.

Para mejorar el proceso de revisiones de forma tal que eleve la calidad del producto final, es necesario tener alguna medida del mejoramiento de las revisiones y de su eficiencia. Para ello se ha determinado una métrica que dotará al proyecto de información necesaria para dar seguimiento al Proceso de Revisiones y recoger las medidas necesarias que contribuyan a mejorarlo.

Se propone la métrica basada en el artículo de la revista investigativa operacional de la Dirección de Información Científica Técnica (DICT) de la Universidad de la Habana: “Una propuesta de introducción de las revisiones en el Proceso de Desarrollo de Software” (Dapena, 2005)

- Efectividad de eliminar los defectos en una Revisión:

$$EED = \frac{DE_i}{DL} * 100$$

$$DL = DE_i + DEP$$

$$DEP = \sum_{k=i+1}^n DE_k$$

DE: cantidad de defectos detectados durante la revisión.

DEP: cantidad de defectos encontrados posterior a la revisión.

DL: cantidad total de defectos presentes en el producto.

Si se representan en un gráfico los valores resultantes de la métrica de efectividad se puede analizar qué revisiones de las realizadas al Proyecto de Software resultan poco efectivas y así valorar la posibilidad de incluirla o no en la misma fase del Desarrollo del Proyecto o en otros cuyas características sean similares al analizado o tomar cualquier otra decisión que contribuya al mejoramiento del Proceso de Revisiones. (Schulmeyer, 1997).

3.3.3 Estándares y guías

Para la elaboración de las distintas actividades que forman el Plan de Aseguramiento de la Calidad del Proyecto Productivo Simulador Quirúrgico se utilizan una serie de estándares y normas que permiten asegurar el proceso de desarrollo de software. (Ver Anexo 5)

3.3.4 Tareas generales de revisiones.

Dentro de las tareas generales se definieron un conjunto de Revisiones Técnicas Formales que dan la posibilidad de encontrar la mayor cantidad posible de defectos y garantizar que el software se realice según los requerimientos propuestos por el cliente.

Revisión Administrativa:

- **Responsable:** Líder del proyecto y Administrador de calidad.
- **Fase en la que se aplica:** Inicio.

- **Artefactos de entrada:** Lista de chequeo para la Revisión Administrativa del Simulador Quirúrgico, Lista de Riesgos del proyecto, Plan de Mitigación de Riesgos, Documento Visión y Propuesta de Roles.
- **Artefactos que genera:** Documentos de No Conformidades, Registro de respuesta a las No Conformidades y Minuta de Reunión.

Esta revisión tiene como objetivo revisar los artefactos creados en la primera fase y su ajuste con las plantillas propuestas en el expediente del proyecto, para ello es preciso confeccionar una lista de chequeo(Ver Anexo 6) teniendo en cuenta los Lineamientos Mínimos de Calidad establecidos por la Dirección de Calidad del Software de la UCI. Las deficiencias encontradas se registran en un documento de No Conformidades y luego el líder del proyecto realiza un plan de respuesta de no conformidades. Esta actividad ayuda a que el proyecto comience correctamente y con una organización lo más consistente posible.

Revisión Post-Morten:

- **Responsable:** Líder del Proyecto y Administrador de la Calidad
- **Fase en la que se aplica:** Al finalizar cada fase.
- **Artefactos de entrada:** Primera versión del producto.
- **Artefactos que genera:** Documentos de No Conformidades, Registro de respuesta a las No Conformidades y Minuta de Reunión.

Tiene como propósito confirmar la aceptación de cada iteración y la aceptación del proyecto y poder brindarles recomendaciones para las iteraciones siguientes. Es responsabilidad del líder del proyecto Simulador Quirúrgico realizar esta actividad al finalizar cada fase.

Revisión a los Requisitos:

- **Responsable:** Revisor Técnico Formal.
- **Fase en la que se aplica:** Inicio y Elaboración.
- **Artefactos de entrada:** Lista de chequeo para la Revisión de los Requisitos en el Simulador Quirúrgico.
- **Artefactos que genera:** Documentos de No Conformidades, Registro de respuesta a las No Conformidades y Minuta de Reunión.

Esta revisión tiene como objetivo revisar los requisitos elaborados, se ejecuta durante la fase de inicio y elaboración, garantizándose que el sistema cumpla con las condiciones o capacidades para satisfacer los requerimientos presentados por el cliente, se confecciona la Lista de chequeo para la Revisión de los Requisitos en el Simulador Quirúrgico (*Ver Anexo 7*) basándose en la Norma de calidad del Instituto de Ingenieros Eléctricos 830 (IEEE 830): “Especificaciones de los requisitos del SW”. En el transcurso en que el revisor técnico va efectuando dicha actividad serán recogidos en un Registro de No Conformidades todos aquellos errores detectados, redactándose posteriormente un registro de respuesta a las No Conformidades.

Revisión a la Arquitectura:

- **Responsable:** Revisor Técnico Formal.
- **Fase en la que se aplica:** Elaboración.
- **Artefactos de entrada:** Lista de chequeo para la Revisión de la Arquitectura en el Simulador Quirúrgico.
- **Artefactos que genera:** Documentos de No Conformidades, Registro de respuesta a las No Conformidades y Minuta de Reunión.

Con esta actividad se desea realizar una revisión detallada a la arquitectura definida en el Simulador Quirúrgico, se confecciona la Lista de chequeo para la Revisión de la Arquitectura en el Simulador Quirúrgico (*Ver Anexo 8*) teniendo en cuenta los Lineamientos Mínimos de Calidad establecidos por la Dirección de Calidad del Software de la UCI y la norma de calidad del Instituto de Ingenieros Eléctricos 1471 (IEEE 1471): “Recomendaciones para la Descripción Arquitectónica de los Sistemas Intensivos de Software”. Se realiza en el flujo de trabajo de Análisis y Diseño de cada una de las iteraciones para cada etapa garantizando un diseño inicial, es realizada por el revisor técnico formal. Al aplicar la lista de chequeo los errores detectados se recogerán en un registro de no conformidades, elaborándose después un registro de respuesta a estas no conformidades encontradas.

Revisión al Diseño:

- **Responsable:** Revisor Técnico Formal.
- **Fase en la que se aplica:** Elaboración.
- **Artefactos de entrada:** Lista de chequeo para la Revisión al Diseño en el Simulador Quirúrgico.

- **Artefactos que genera:** Documentos de No Conformidades, Registro de respuesta a las No Conformidades y Minuta de Reunión.

Con esta actividad se desea detectar todos los errores existentes antes de pasar a la fase de implementación. Se realizará a partir de la Lista de chequeo para la Revisión al Diseño en el Simulador Quirúrgico (*Ver Anexo 9*) elaborada teniendo en cuenta los Lineamientos Mínimos de Calidad establecidos por la Dirección de Calidad del Software de la UCI y la Norma de calidad del Instituto de Ingenieros Eléctricos 1016(IEEE 1016): “Prácticas recomendables para la descripción del diseño del software” y la ISO 12207:”Tecnología de la Información. Procesos del ciclo de vida del software”. Esta actividad se debe aplicar durante la fase de Elaboración a cada iteración de cada una de las etapas del Simulador Quirúrgico y su responsable es el revisor técnico formal. En el proceso de ejecución se recogerán las no deficiencias y se registrarán en el registro de no conformidades, continuamente se realizará un registro de respuestas a las no conformidades encontradas.

Revisión a la documentación del proyecto:

- **Responsable:** Revisor Técnico Formal.
- **Fase en la que se aplica:** Al finalizar cada etapa del proyecto.
- **Artefactos de entrada** Guía para la revisión de la documentación. Dirección de Calidad del Software.
- **Artefactos que genera:** Documentos de No Conformidades, Registro de respuesta a las No Conformidades y Minuta de Reunión.

Esta actividad tiene como objetivo revisar toda la documentación que se genera en el proyecto Simulador Quirúrgico dándole especial atención a los documentos que conforman el expediente del proyecto. Tiene como responsable al Revisor Técnico Formal y para su ejecución se elabora la lista de chequeo para la Revisión a la Documentación (*Ver Anexo 10*) teniendo en cuenta los Lineamientos Mínimos de Calidad establecidos por la Dirección de Calidad del Software de la UCI y la Guía para la revisión de la documentación. Esta actividad se realizará al finalizar cada etapa del proyecto. Al encontrar alguna deficiencia será recogida en el documento de no conformidades y luego es elaborado un Registro de respuesta a las no conformidades detectadas.

Revisión a la Gestión de Configuración:

- **Responsable:** Revisor Técnico Formal.
- **Fase en la que se aplica:** Antes de incluir la línea base del proyecto en el repositorio de elementos de configuración.
- **Artefactos de entrada:** Lista de chequeo para la Revisión a la Gestión de Configuración en el Simulador Quirúrgico.
- **Artefactos que genera:** Documentos de No Conformidades y Registro de respuesta a las No Conformidades.

Se desea comprobar que se están controlando los cambios, que la documentación y el expediente del proyecto se encuentren y a su vez estén actualizados. Para comprobar esto se realizó la lista de chequeo para la Revisión a la Gestión de Configuración en el Simulador Quirúrgico (Ver Anexo 11) en correspondencia con los Lineamientos Mínimos de Calidad establecidos por la Dirección de Calidad del Software de la UCI y la norma ISO 12207: "Tecnología de la Información. Procesos del ciclo de vida del Software". Se realiza antes de incluir la línea base del proyecto en el repositorio de elementos de configuración, dando la posibilidad a los elementos de configuración de mantener la estabilidad en cada una de las iteraciones para las etapas del Simulador Quirúrgico. Su responsable es el revisor técnico formal, al aparecer una deficiencia se archivará en el documento de no conformidades y seguidamente se redacta el registro de respuesta a las no conformidades.

3.3.5 Pruebas

Pruebas de Unidad: Es realizada con el objetivo de comprobar la calidad del código de cada uno de los módulos o componentes partiendo del diseño de los casos de pruebas. En ella se descubren la mayoría de los errores de programación durante la propia etapa, mientras más tiempo permanezca un error en el sistema, más tiempo requerirá eliminarlo y tendrá mayor repercusión en otras secciones del programa. Para estas pruebas se utilizará la técnica de:

Prueba de Caja Blanca.

- **Responsable:** Equipo de desarrolladores.
- **Fase en la que se aplica:** Elaboración y Construcción.

- **Artefactos de entrada:** Diseño de Casos de Pruebas de Caja Blanca para el Simulador Quirúrgico
- **Artefactos que genera:** Registro de No Conformidades y Registro de Respuesta a No Conformidades.

Estas son realizadas con el objetivo de comprobar la calidad del código, se realiza durante la última iteración de la fase de elaboración y en la fase de construcción. Es ejecutada a partir de los casos de pruebas diseñados con anterioridad guiándose por las normas ISO 12207: Tecnología de la Información. Procesos del ciclo de vida del Software, la ISO 9126: Tecnología de la información. Características de Calidad y Métricas del Software y la ISO 12119: El Software, Tecnología de la Información, Requisitos de Calidad y Pruebas”. El responsable de la realización de esta actividad es el equipo de desarrolladores. Las deficiencias que se encuentren en el transcurso de estas pruebas serán recogidas en un registro de No Conformidades y se realizará un registro de respuesta a las no conformidades encontradas por parte de los desarrolladores del módulo implicado.

Pruebas de Integración: Esta actividad se realiza con el objetivo de comprobar que la integración de los componentes sea de la manera correcta. Para su realización se utilizará la estrategia de:

Integración Ascendente:

- **Responsable:** Probador.
- **Fase en la que se aplica:** Terminada la primera iteración de la fase de Construcción.
- **Artefactos de entrada:** Diseño de Casos de Pruebas de Integración Ascendente para el Simulador Quirúrgico.
- **Artefactos que genera:** Registro de No Conformidades y Registro de Respuesta a No Conformidades.

Con esta prueba se tiene como objetivo la integración de los componentes de los niveles más bajos es decir una integración de abajo hacia arriba. Esta actividad se realiza en la fase de construcción. El responsable de su realización es el probador y se efectúa teniendo en cuenta el diseño de los casos de pruebas anteriormente elaboradas. Se va combinando un componente con el siguiente componente que debe

ser probado. Los defectos detectados se registran en el documento de no conformidades y se elabora un registro de respuesta a las no conformidades.

Prueba de Sistema: Con la realización de esta prueba se desea verificar el programa final después de que todos los componentes han sido integrados. Para su realización se utilizarán las pruebas de caja negra:

Pruebas de Caja Negra:

- **Responsable:** Probador.
- **Fase en la que se aplica:** Construcción.
- **Artefactos de entrada:** Diseño de Casos de Pruebas de Caja Negra para el Simulador Quirúrgico.
- **Artefactos que genera:** Registro de No Conformidades y Registro de Respuesta a las No Conformidades.

Se pretende comprobar la funcionalidad del producto, esta prueba será efectuada en la última iteración de la fase de construcción y se realizará a partir del Diseño de casos de pruebas de caja negra, para la realización de este diseño de casos de prueba se tendrá en cuenta la Norma de calidad del Instituto de Ingenieros Eléctricos 1028 (IEEE 1028):“Estándar para la revisión de Software” y la norma ISO 9126: Tecnología de la información. Características de Calidad y Métricas del Software. Las deficiencias encontradas se recogerán en el registro de no conformidades y se realizará un registro de respuesta a las no conformidades. Estas pruebas tienen como responsable al Probador.

3.3.6 Herramientas, técnicas y metodologías

Las herramientas, técnicas y metodologías que apoyan el plan de aseguramiento de la calidad se nombran a continuación:

- Se utiliza la metodología RUP como guía en la ejecución de las pruebas, el plan de prueba y el diseño de los casos de prueba.
- Se aplican los Lineamientos Mínimos de Calidad definidos por la Dirección de Calidad del Software de la UCI.
- Se utilizan las Plantillas del Expediente de Proyecto para registrar la documentación del proyecto.

- Se aplican las Listas de Chequeo como herramienta a utilizar en el desarrollo de las revisiones.
- Se emplean las Revisiones Técnicas Formales como técnica para revisar los artefactos producidos por la ingeniería de software.

3.3.7 Resolución de problema y acción correctiva

Al detectar cualquier error, problema o incongruencia en cualquiera de las etapas de revisión y pruebas, se levantará un documento titulado Registro de No Conformidades (*Ver Anexo 12*), donde será relacionando la no conformidad encontrada. Este documento se le hará llegar al líder del proyecto, una copia al responsable de mitigar o eliminar el problema, y otra a la documentación manejada dentro del equipo de calidad.

El líder del proyecto en conjunto con el jefe de equipo del área sometida a revisión o prueba elaborarán un documento denominado Registro de Respuesta a las No Conformidades (*Ver Anexo 13*) que será entregado al grupo interno de calidad del proyecto y al responsable de eliminar la no conformidad, luego se producirá un ciclo de revisiones entre el grupo interno de calidad del proyecto y el equipo de desarrolladores, hasta darle solución a la No conformidad.

3.3.8 Registros de calidad

El equipo de calidad tiene establecido como política a seguir que toda la documentación de calidad que se genere sea conservada durante todo el ciclo de vida del proyecto e incluso después de concluido pues constituye una evidencia del trabajo realizado y una fuente de consulta muy valiosa que se puede utilizar en otros proyectos productivos de la Facultad o Universidad.

Los registros de calidad que se guardarán serán los siguientes:

- ❖ Listas de Chequeo Aplicadas.
- ❖ Casos de Prueba.
- ❖ Actas de Reuniones Efectuadas.
- ❖ Los informes de No Conformidades.
- ❖ El Registro de Respuesta a las No Conformidades.

Todos estos registros serán debidamente guardados en el Repositorio del Expediente del proyecto.

3.4 Proceso de Revisiones en el Simulador Quirúrgico

El proceso de revisiones en el Simulador Quirúrgico comienza cuando el equipo de aseguramiento de la calidad precisa la planificación, en que flujo de trabajo se ejecutará, las herramientas y la labor de los roles en las revisiones. Los roles encargados del proceso de revisiones son el administrador de la calidad y el revisor técnico formal.

Para la realización de las revisiones primeramente hay que planificarlas para darle seguimiento a las actividades de Ingeniería del Software con el objetivo de que se le de cumplimiento a los requerimientos planteados por el cliente, luego de una buena planificación se establecen las herramientas a utilizar para la realización de las revisiones, en este caso son las Listas de Chequeo. Cuando el artefacto a revisar no se encuentra listo para ser revisado se le aplica la revisión a lo que hay hecho hasta el momento y en el registro de no conformidades se especifica detalladamente los motivos que lo provocan y se le pone una nueva fecha para volver a revisar completamente. En el caso de que el artefacto se encuentre en condiciones de ser revisado el administrador de la calidad es el encargado de informarles al jefe de proyecto, revisor técnico formal y a todas aquellas personas involucradas de una manera u otra en la actividad la fecha de su realización.

El artefacto a revisar debe ser enviado con dos días de antelación para que los participantes se reúnan inicialmente realizándole una revisión individual al artefacto utilizando la lista de chequeo definida con anterioridad y las no conformidades detectadas se informarán en la reunión de revisión.

Ya en la reunión de revisión se discuten las no conformidades y se plasman en el documento de No Conformidades definido por la Dirección de Calidad del Software de la UCI. En esta reunión de revisión también es redactado un documento de Minuta de Reuniones en el que quedarán plasmados los participantes y un resumen de los acuerdos tomados. Luego de haber culminado la realización de la revisión es de suma importancia darle el seguimiento correspondiente a las no conformidades detectadas siendo responsable de esta actividad el líder del proyecto en conjunto con el responsable del área que esta siendo sometida a revisión los que realizarán un registro de respuestas a las no conformidades el que se le entregará al encargado de mitigarlos, sometiéndolo nuevamente a una revisión, se realiza un ciclo de revisiones al artefacto hasta que se hayan mitigado todas las no conformidades encontradas.

El administrador de la calidad es el encargado de aplicar la métrica efectividad de eliminar los defectos en una revisión al culminar la revisión proporcionando una vista global de la eficacia de la revisión.

Aún cuando los resultados son satisfactorios las revisiones ayudan a ganar experiencia y de esta manera se puede analizar dónde puedan estar los problemas y utilizar la información resultante de las revisiones para el mejoramiento del proceso de desarrollo del software.

3.5 Proceso de pruebas en el Simulador Quirúrgico

El proceso de pruebas en el Simulador Quirúrgico comienza con la confección del Plan de Pruebas, en este plan se describe de forma específica las estrategias, recursos y planificación de las pruebas a realizar. En el Plan de pruebas queda definido la organización del equipo de prueba como se muestra a continuación: administrador de la calidad, diseñador de casos de pruebas y el probador, en el caso de las pruebas de caja blanca el probador debe ser un desarrollador que tenga conocimiento de implementación.

Al planificarse las pruebas se tienen en cuenta el software y hardware al que se recurre incluyendo los proveedores y sus versiones. También se realiza una descripción de los requisitos a probar dentro de las pruebas establecidas, especificando que solo se le realizarán las pruebas a los requisitos que sean mencionados. En este plan se hace una reseña de los posibles casos de pruebas y los métodos que se utilizarán para ello, así como también del proceso de ejecución y evaluación de las pruebas. Se realizarán pruebas de integración, unidad y sistema, las que se encuentran plasmadas en el Plan de pruebas del Simulador Quirúrgico.

Luego de planificarse las pruebas en el Plan de pruebas los diseñadores de casos de pruebas se prepararán para diseñar los casos de pruebas. En un principio los diseñadores de casos de pruebas evalúan la documentación necesaria para la realización de los casos de pruebas entregada por los desarrolladores. Con esta información en sus manos los diseñadores de casos de pruebas definen el objetivo de las pruebas, los métodos a utilizar para el diseño y se elaboran las listas de chequeo de comprobación. Al concretar todo lo anterior el equipo de diseñadores se encuentra listo para comenzar a diseñar las pruebas registrándolas en el documento de Diseño de Casos de pruebas definido por la Dirección de Calidad del Software de la UCI.

Una vez diseñado los casos de pruebas correspondientes se comenzarán a ejecutar las pruebas cuando se encuentren preparados los recursos tanto humanos como

materiales y el producto que será objeto de las pruebas. Si alguno de estos elementos no está listo y ha llegado la fecha planificada para la realización de las pruebas se plasma en el documento de no conformidades, explicando las causas y se ponen de acuerdo con los desarrolladores sobre cuando el producto estará listo, planificando una nueva fecha. Si el producto está listo para realizarle la prueba el administrador de calidad es el responsable de asignar las tareas a los probadores y se comienza a ejecutar. Conjuntamente a la ejecución de las pruebas se irán comparando los resultados que se obtienen con los previstos en el diseño de las pruebas, si se revelan diferencias estas se registran en el documento de no conformidades definido por la Dirección de Calidad del Software de la UCI, estas no conformidades deben ser entregadas a los desarrolladores lo más rápido posible para que las analicen y corrijan, y su vez se realiza el registro de respuesta a los no conformidades detectadas.

Al concluir cada iteración se deben valorar los resultados alcanzado con las pruebas, analizar la tendencia de las no conformidades detectadas y la rapidez en darle solución, todo esto será incluido en el informe final de las pruebas. En este informe se incluirá además las lecciones aprendidas con el proceso de prueba realizado lo cual servirá de referencia para próximas pruebas.

3.6 Validación de la estrategia

Teniendo en cuenta que el objetivo de este trabajo es el de proponer una estrategia de aseguramiento de la calidad para el Simulador Quirúrgico, esta debe ser evaluada por un personal con alta capacitación sobre el tema.

Con este fin se utiliza el Método Delphi, método cualitativo que procede por medio de la interrogación a expertos con la ayuda de cuestionarios a la evaluación de un proyecto que se pondrá en práctica o alguna propuesta de desarrollo. La encuesta se lleva a cabo de manera anónima (actualmente es habitual realizarla haciendo uso del correo electrónico) para evitar los efectos de “líderes” (Caseres, 2006).

A continuación se hará la descripción de las técnicas que fueron utilizadas del método y los resultados obtenidos.

- Selección de expertos

¿Qué significa experto?

Se entiende por experto, tanto al individuo en sí, como a un grupo de personas u organizaciones capaces de ofrecer valoración conclusiva de un problema en cuestión y hacer recomendaciones respecto a sus momentos fundamentales con un máximo de competencia (Caseres, 2006).

Para la selección de los expertos, se tomaron en cuenta las áreas de conocimientos que están asociadas a la solución de la estrategia propuesta y que atributan en la conformación de la estrategia de manera directa o indirecta, ya que aportan elementos y conceptos claves para su elaboración. Las áreas que fueron definidas y que tienen relación con el procedimiento a evaluar son:

- ✓ Calidad y Gestión del Software.
- ✓ Proceso de Aseguramiento de la Calidad del software.
- ✓ Normas y estándares internacionales de calidad.

Todos los posibles candidatos tenían conocimiento general de todos los temas abordados en las áreas de conocimiento que abarca la estrategia propuesta. En total se escogieron 7 expertos integrantes de la Universidad de las Ciencias Informáticas.

- Consentimiento del experto para su colaboración

Una vez que el listado fue confeccionado, se realizó una invitación personal a cada uno de los expertos escogidos de forma personal para que participaran en la actividad. Se le explicó el objetivo de la realización de la encuesta, el plazo y su orden de ejecución, así como el volumen total del trabajo en general. Una vez que fue recibida la respuesta positiva, se estableció el listado final de expertos, de lo cual se le informó a cada especialista sobre su inclusión en la actividad y las instrucciones necesarias para contestar las preguntas. Con esto culminó el trabajo de selección lográndose la participación de los 7 expertos escogidos.

- Elaboración de la encuesta a desarrollar

Para la confección de la encuesta a desarrollar se tuvieron en cuenta los principios básicos que debería cumplir la estrategia para su puesta en práctica en el proyecto productivo Simulador Quirúrgico, además de los elementos que existen dentro de la organización en la actualidad y que podrían imposibilitar su total desarrollo. La encuesta fue conformada con preguntas cerradas, sobre la validez de la estrategia

planteada y la evaluación del proceso, cerrando el entorno de las respuestas a los puntos básicos del problema tratado durante el desarrollo del trabajo (Ver Anexo 14).

En todos los casos los expertos recibieron la documentación de la estrategia propuesta y se les requirió cumplir con un plazo de tiempo determinado para dar las respuestas o hacer las preguntas pertinentes que les hubiesen surgido al estudiar el documento presentado.

- Resultados de la valoración de los expertos

Después de ser procesada la encuesta se pudo determinar que todos los expertos estuvieron de acuerdo con la estrategia de aseguramiento de la calidad desarrollada en este trabajo de diploma para el proyecto Simulador Quirúrgico.

Las referencias de por qué estas respuestas, prevalecieron en que de esta manera se podía alcanzar en el proyecto procesos y productos de mayor calidad, mejor elaboración y con la documentación debidamente realizada, como lo sugieren las buenas prácticas de Aseguramiento de la Calidad del Software.

En las encuestas se trabajó sobre cuáles eran los principios por los que los expertos podían evaluar la estrategia de aseguramiento de la calidad para el Simulador Quirúrgico. En las respuestas dadas por los expertos hubo variabilidad como se muestra a continuación (Ver Tabla 3.1).

	Exp1	Exp2	Exp3	Exp4	Exp5	Exp6	Exp7
¿En qué medida considera UD correcto el desarrollo de la estrategia propuesta?	Alta	Alta	Alta	Alta	Alta	Alta	Alta
¿En qué nivel los roles y responsabilidades propuestas pueden lograr eficiencia y máxima calidad en el proyecto Simulador Quirúrgico?	Alta	Alta	Alta	Alta	Alta	Alta	Alta
¿En qué medida considera UD correcta las condiciones necesarias	Alta	Media	Alta	Alta	Media	Alta	Alta

que se describen para que la investigación sea desarrollada con éxito?							
¿En qué nivel considera UD correcta las actividades descritas en la estrategia?	Alta						

Tabla 3.1: Resultados de las preguntas realizadas a los expertos.

Como se puede observar, todos los expertos consideran en su evaluación que el desarrollo de la estrategia propuesta es correcto en alta medida, al igual que las actividades descritas en la estrategia. La capacidad de los roles y responsabilidades propuestos para lograr eficiencia y máxima calidad en el proyecto también recibieron una evaluación de alta por parte de la totalidad de los expertos.

Pero sin embargo algunos expertos consideran que las condiciones necesarias que se describen en la investigación para que esta sea desarrollada con éxito es correcto en un nivel medio.

Se puede llegar a la conclusión que la estrategia de aseguramiento de la calidad para el Simulador Quirúrgico propuesta en el presente trabajo de diploma tuvo una validación por parte de los expertos con un alto porcentaje de aceptación. (Ver Figura 3.1)

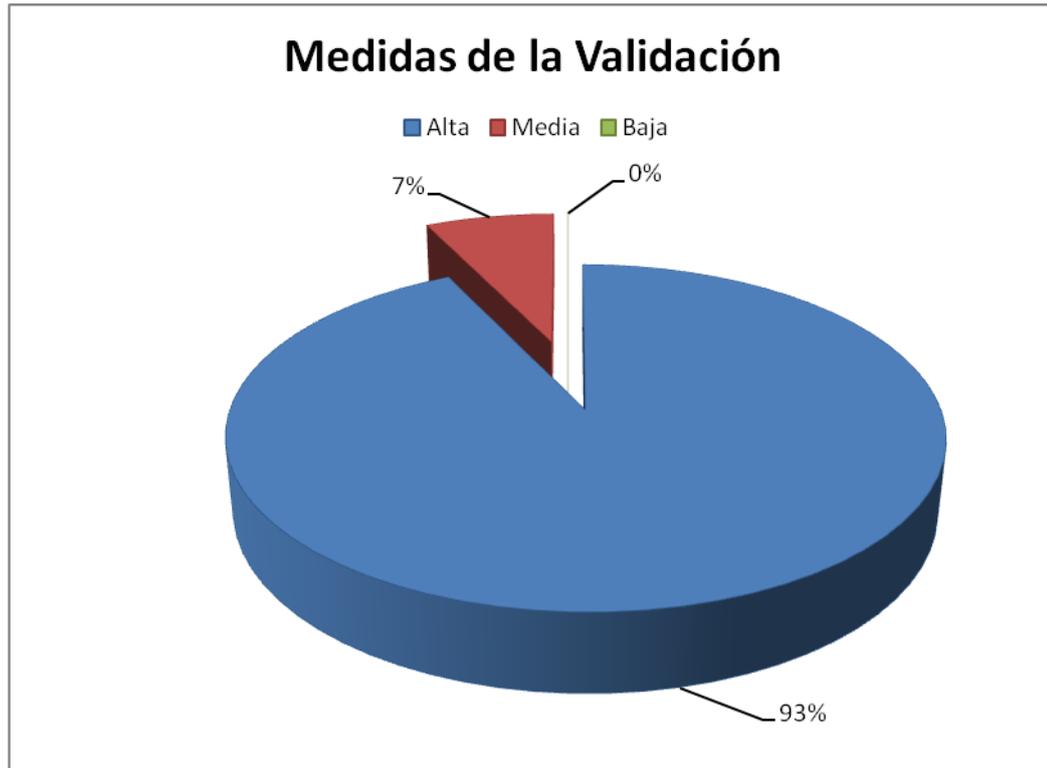


Figura 3.1: Porciento total de la medida de Evaluación.

Esta evaluación fue dada en esa medida debido a que ayuda a asegurar que el desarrollo de los procesos y productos realizados en el Simulador Quirúrgico obtengan un nivel de calidad pues se efectúa una planificación adecuada del proceso de aseguramiento de la calidad así como de las actividades que se ejecutarán en el trayecto en que transcurra el proyecto para mejorar las actividades de ingeniería del software y darle cumplimiento a los requerimientos propuesto por el cliente en un inicio.

CONCLUSIONES

Después de elaborada la Estrategia de Aseguramiento de la Calidad para el proyecto productivo Simulador Quirúrgico y de ser evaluada por el criterio de un grupo de expertos, se llegaron a las siguientes conclusiones:

- Se estudiaron los diferentes enfoques de aseguramiento de la calidad según algunos de los modelos, estándares y normas para el desarrollo del software.
- Se realizó un estudio detallado del proceso actual de aseguramiento de la calidad en el Simulador Quirúrgico especificándose los señalamientos detectados en el proyecto en una auditoría realizada por la dirección de calidad de la universidad.
- Se definió una estrategia de aseguramiento de la calidad para el proyecto Simulador Quirúrgico sustentada por el Plan de aseguramiento de la calidad como artefacto fundamental, además de la definición del equipo de calidad, y los procesos de revisiones y pruebas en el proyecto.
- Se validó la estrategia definida a partir de la aplicación del método Delphi obteniéndose resultados satisfactorios aceptándose la aplicación de la propuesta a partir de la segunda etapa de desarrollo del proyecto.

RECOMENDACIONES

- Dar seguimiento a las actividades del Plan de Aseguramiento de la Calidad durante todo el ciclo de vida del proyecto Simulador Quirúrgico.
- Cumplir con el Plan de Pruebas durante todo el ciclo de desarrollo.
- Desarrollar la métrica propuesta en este trabajo así como analizar y estudiar otras métricas que sean de posible y fácil aplicación en el proyecto.

REFERENCIA BIBLIOGRÁFICA

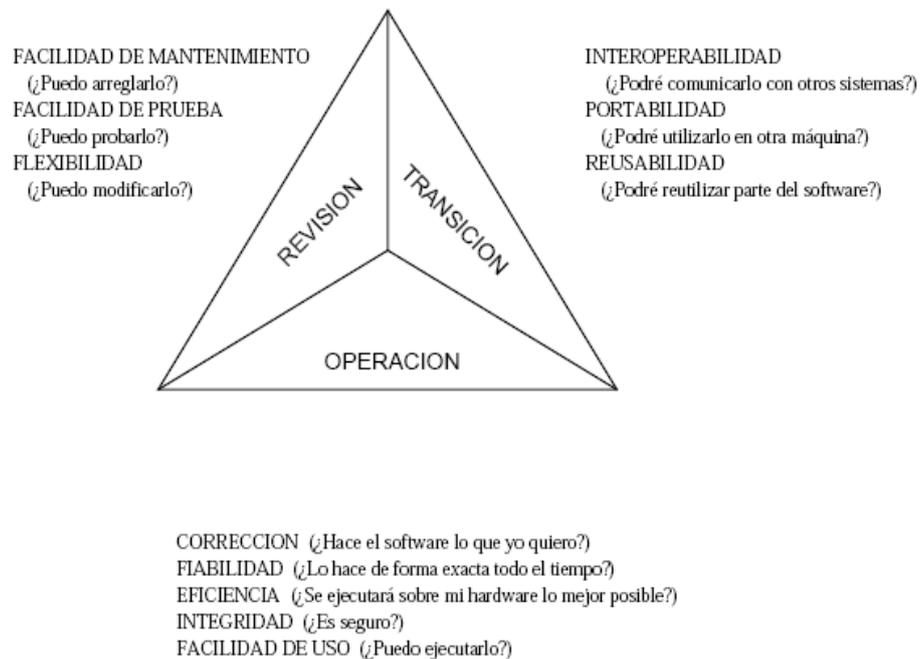
1. **Andújar U, Juan Antonio. 2004.** Métricas de Calidad para Proyectos de Software. 2004.
<http://www.pmies.org/ponencias/2004/ponencia%20Juan%20A%20Andujar.pdf>..
2. **Antonio, A. D. 2002.** *Gestión, Control y Garantía de la Calidad de Software.* 2002.
3. http://www.inf.uach.cl/rvega/asignaturas/info265/G_Calidad.pdf
4. **Baeza, Yates R. A. y otros. 1995.** *Computing in Chile: The Jaguar of the Pacific Rim Communications of the ACM.* 1995. Vol. 38.
5. **Bañeres, J.P.** *Introducción a ISO/IEC 15504.*
6. **Buades, Rubio Gabriel. 2002.** *Calidad en ingeniería del Software.* 2002.
7. **Caseres, E. 2006.** El método Delphi. Características. 2006.
<http://www.codesyntax.com>.
8. **Carrasco, F. y O. M, et al. 2004.** *Un enfoque actual sobre la calidad del software.* 2004.
9. **CMMI. 2006.** *CMMI for Development, Version 1.2.* 2006.
10. **Control, Departamento de y Auditoría Informática. 2000.** Control de calidad en los servicios. 2000.
<http://sistemas.dgsca.unam.mx/publica/pdf/Control%20de%20Calidad.PDF> .
11. **Dapena, Delgado Martha Dunia y otros. 2005.** *Una Propuesta de Introduccion de las Revisiones en el Proceso de Desarrollo de Softwre.* Ciudad de La Habana : Instituto Superior Politecnico "José Antonio Echeverría" : s.n., 2005.
12. **Deming, Edward. 1989.** *"Calidad, Productividad y Competitividad"*. 1989.
13. **Escalona, María J y Koch, Nora. 2002.** Ingeniería de Requisitos en Aplicaciones para la Web. Un estudio comparativo. 2002.
<http://lsiweb.lsi.us.es/docs/informes/LSI-2002-4.pdf>..
14. **Febles, Estrada Ailyn. 2000.** *Calidad de Software y la empresa, enseñanza de un tema imprescindible para el Ingeniero Informático.* Instituto Superior Politécnico "José Antonio Echeverría" : s.n., 2000.
15. **Ferré, Xavie rGrau. 2005.** *Marco de Integración de la Usabilidad en el Proceso de Desarrollo de Software.* 2005.

16. **Furone, Francisco. 1998.** Realidad Virtual. 1998.
<http://www.monografias.com/trabajos/vr/vr.shtml?monosearch>.
17. **Goodman, Paul. 2004.** Software Metrics-Best Practices for Successful IT Managemen. 2004. <http://sharespot.flazx.net/download/1931332266.zip.htm>. ISBN: 1- 931332-26-6..
18. **Hilera, José R., Salvador Otón, Javier Martínez. 2000.** Aplicación de la Realidad Virtual en la enseñanza a través de Internet. 2000.
<http://www.ucm.es/info/multidoc/multidoc/revista/num8/hilera-oton.html> ..
19. **Espasa, -Calpe S.A. 2005.** Diccionario de la lengua española. 2005.
<http://www.wordreference.com/definicion/simulador>.
20. **IEEE. 1990.** *IEEE Standard Glossary of Software Engineering Terminology.* 1990.
21. **ISO 9000. 2000.** "Sistema de Gestión de la Calidad". 2000.
http://fabetsia.dmpa.upm.es/solo_alumnos/sp2/Tablon_sp2/TransparenciasCALI DAD06.pdf.
22. **Jacobson, Ivar, Booch, Grady y Rumbaugh, James. 2000.** *El Proceso Unificado de Desarrollo de Software.* 2000.
23. **Juran, Dr Joseph M. 1979.** <http://www.juran.com/>..
24. **Juran, Joseph M y Gryna, F. 1993.** "Manual de control de la calidad". 4ta . s.l. : Mc Graw-Hill, 1993.
25. **Mendoza, Luis Eduardo. 2005.** Sistemas de Información III. 2005.
[http://prof.usb.ve/lmendoza/Documentos/PS-6117%20\(Teor%EDa\)/PS6117%20Calidad%20del%20Software.pdf](http://prof.usb.ve/lmendoza/Documentos/PS-6117%20(Teor%EDa)/PS6117%20Calidad%20del%20Software.pdf).
26. **Navarro, Jose Luis Tomás y Sergio Pérez Paredes. 2003.** Estandar ISO 9000-3. 2003.
<http://www.dsic.upv.es/asignaturas/facultad/lsi/trabajos/092000.ppt#256,1,ESTANDAR>
27. **Pressman, Roger S. 2005 .** *Ingeniería del Software, Un enfoque práctico.* La Habana : Félix Varela, 2005 .
28. **RUP. 2003.** Rational Unified Process. 2003.

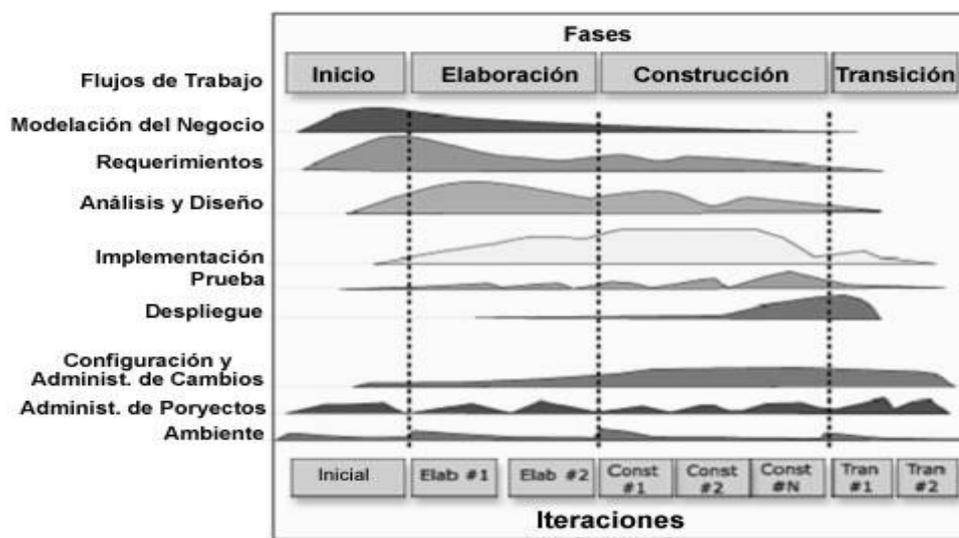
29. **Scalone. 2006.** Estudio comparativo de los modelos y estándares de calidad del software. 2006. <http://www.fi.uba.ar/laboratorios/lsi/scalone-tesis-maestria-ingenieria-en-calidad.PDF..>
30. **Schulmeyer, G. 1997.** "Handbook of Software Quality Assurance". 1997.
31. **Toiran, Y. 2006.** Organización en Procesos del Departamento de Detención a Estudiantes Extranjeros. 2006.
32. **Torregrosa, Rafael Sánchez.** Calidad concepto y generalidades. http://chguv.san.gva.es/Descargas/Gerencia/PlaniCalidad/Kiosco/Calidad_generalidades.pdf.

ANEXOS

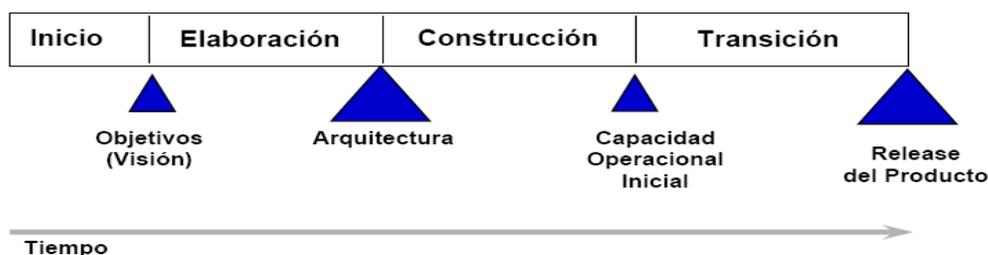
Anexo1 Factores MacKall



Anexo 2 Fases, Iteraciones y Flujos de Trabajo de RUP



Anexo 3 Fases e Hitos



Anexo 4 Áreas de Desarrollo del Modelo CMMI

Nivel de Madurez de la Organización	Centrado en	Áreas de Procesos
1. Inicial	Proc. impredecible, control reactivo.	
2. Gestionado	Gestión básica del proyecto.	<ul style="list-style-type: none"> - Gestión de requisitos. -Planificación de proyectos. -Monitorización y control de proyectos -Gestión de acuerdos con proveedores. -Aseguramiento de la calidad del producto y del proceso -Gestión de la configuración.
3. Definido	Proceso caracterizado por la organización y proactivo.	<ul style="list-style-type: none"> -Desarrollo de los requisitos -Soluciones técnicas -Integración de productos -Verificación -Validación - Desarrollo y mejora de procesos en la organización

		<ul style="list-style-type: none"> -Definición de procesos en organización. -Gestión del riesgo -Análisis y resolución de toma de decisiones -Planificación de la formación.
4. Gestionado Cuantitativamente	Control cuantitativo del proceso	<ul style="list-style-type: none"> -Gestión cuantitativa de los proyectos. -Mejora de los procesos de la organización.
5. Optimizado	Mejora continua del proceso	<ul style="list-style-type: none"> -Análisis y resolución de causas de desviaciones. -Innovación y despliegue a toda la organización.

Anexo 5 Estándares y Normas utilizados.

Estándar	Nombre	Descripción
LMC	Lineamientos Mínimos de Calidad	
CMMI	Integración del Modelo de Capacidad y Madurez.	El propósito de CMMI para el desarrollo es ayudar a las organizaciones a mejorar sus procesos de desarrollo y mantenimiento de los productos y los servicios.
ISO 9001:2000	Sistema de Gestión de la Calidad. Requisitos.	Promueve la adopción de un enfoque basado en procesos cuando se desarrolla, implementa y mejora la eficacia de un sistema de gestión de la calidad, para aumentar la satisfacción del cliente mediante el cumplimiento de sus requisitos.
ISO 9126	Métricas de Software.	Permite especificar y evaluar la calidad del producto de software desde perspectivas

		<p>diferentes de aquellos asociados con la adquisición, regulación, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad y auditoría del software basado en los atributos internos del software y el comportamiento externo del sistema. Estos tipos de métricas son aplicables al especificar requisitos de calidad y objetivos del diseño para los productos del software, incluso los productos intermedios.</p>
ISO 12207	<p>Procesos del ciclo de vida del software.</p>	<p>Establece una estructura común para los procesos del ciclo de vida del software, con una terminología bien definida, que puede ser tomada como referencia por la industria del software. Contiene procesos, actividades y tareas que se aplican durante la compra de un sistema que contiene software, un “producto de software solamente” (stand-alone software product), un servicio de software y durante el suministro, desarrollo, operación y mantenimiento de productos de software. El software incluye la porción de software del “firmware”.</p> <p>Esta Norma Internacional también proporciona un proceso que puede emplearse para definir, controlar y mejorar procesos del ciclo de la vida del software.</p>
ISO 9003:2004	<p>Guía para la aplicación de la norma ISO 9000:2000</p>	<p>Este estándar internacional proporciona la dirección para las organizaciones en el uso</p>

	al software de computación.	del 9001:2000 de la ISO a la adquisición, a la fuente, al desarrollo, a la operación y al mantenimiento de los programas informáticos y de los servicios de asistencia relacionados.
IEEE 730	Estándar para los planes de aseguramiento de la calidad del software.	El propósito de este estándar es proporcionar uniformidad, los requisitos mínimos aceptables para la preparación del contenido de los planes de aseguramiento de la calidad del software (SQAPs).
IEEE 829	Estándar para la documentación de las pruebas del software.	El propósito de este estándar es describir un sistema de documentos básicos de las pruebas del software.
IEEE 830	Especificaciones de los requisitos del software.	En este estándar se describen criterios que se recomiendan para la especificación de los requisitos del software. Está basado en un modelo donde el resultado del proceso de especificación de los requerimientos del software es un documento de especificación inequívoco y completo.
IEEE 1016	Prácticas para la descripción del diseño de software.	Este estándar especifica la información necesaria contenida y recomienda una organización para la descripción del diseño del software. Este documento no apoya explícitamente, ni es limitado para una metodología de diseño de software o una tecnología descriptiva. Este guiará la producción de documentos de diseño hasta información de diseño para una base de datos automatizada. Para una organización el proceso de desarrollar una

		<p>descripción estándar de diseño, el uso de este documento será de ayuda para este nuevo estándar, encontrando las necesidades para todos los usuarios. Para una organización con un estándar maduro de diseño debe probar en la evaluación y modificación, ese estándar en crecimiento es para las necesidades informativas y de organización necesario para la descripción de diseño de la comunidad de usuario.</p>
IEEE 1471	<p>Práctica recomendada para la descripción arquitectónica de sistemas Software-Intensivos</p>	<p>El objetivo de este estándar es el de facilitar la expresión y la comunicación de la arquitectura y por tanto poner las bases para la calidad y el beneficio del costo por la estandarización de elementos y prácticas para la descripción arquitectónica.</p>

Anexo 6 Lista de Chequeo para la Revisión Administrativa del Simulador Quirúrgico.

	Criterio de Evaluación	S	N	NP	Observación
1	¿Se encuentra elaborado el documento visión?				
2	¿Está elaborado utilizando la plantilla propuesta en el Expediente del Proyecto?				
3	¿Están completamente documentados todos los aspectos de la plantilla?				
4	¿Se discutió la propuesta de visión del proyecto con todos los miembros del equipo de desarrollo?				

5	¿Están definidos los riesgos del proyecto?				
6	¿Está elaborado utilizando la plantilla propuesta en el Expediente del Proyecto?				
7	¿Están completamente documentados todos los aspectos de la plantilla?				
8	¿Se encuentra elaborado el Plan de Mitigación de Riesgos del proyecto?				
9	¿Está elaborado utilizando la plantilla propuesta en el Expediente del Proyecto?				
10	¿Están completamente documentados todos los aspectos de la plantilla?				
11	¿Se encuentran definidos todos los roles necesarios para el proyecto?				
12	¿Se definen las responsabilidades a cada uno de los roles definidos?				

Anexo 7 Lista de Chequeo para la Revisión de los Requisitos en el Simulador

Quirúrgico.

	Criterio de Evaluación	S	N	NP	Observación
1	¿El requisito tiene más de una interpretación?				
2	¿El requisito está definido de forma clara para que el usuario o cliente lo pueda entender?				
3	¿El requisito está delimitado en términos cuantitativos?				
4	¿Esta definida y documentada una metodología específica para la captura de requisitos?				
5	¿Es correcto el requisito? o ¿El requisito es relevante al problema y a su solución?				
6	¿Es inequívoco el requisito?				

7	¿Es completo el requisito? o ¿Existen requisitos que sean imposibles de cumplir?				
8	¿El requisito es consistente?				
9	¿El requisito es verificado y validado?				
10	¿El requisito es modificable?				
11	¿El requisito incumple alguna restricción definida?				
12	¿Se puede seguir el requisito en el modelo del sistema que hemos desarrollado?				
13	¿Se puede localizar el requisito en el conjunto de objetivos del sistema/producto?				
14	¿Están los requisitos asociados con los rendimientos del sistema o con su comportamiento y han sido establecidas claramente sus características operacionales? ¿El requisito está implícitamente definido?				
15	¿Cada requisito tiene asignado un nivel de prioridad asignado por el Cliente/Usuario para guiar las negociaciones/compromisos?				
16	¿Se construyen prototipos de interfaz de usuario para capturar requisitos?				

Anexo 8 Lista de chequeo para la Revisión de la Arquitectura en el Simulador Quirúrgico.

	Criterio de Evaluación	S	N	NP	Observación
1	¿Se priorizan los casos de usos significativos para la arquitectura?				
2	Se verifica que el modelo de arquitectura cumple con los objetivos acordados.				

3	¿Se realizan estudios que permitan identificar las herramientas adecuadas para el desarrollo de software en cuestión. ?				
4	¿Se Identifican patrones de Arquitectura?				
5	¿Se documenta la información referente a la descripción de la arquitectura?				
6	Se identifican los stakeholders del sistema				
7	¿Se identifican las preocupaciones y opiniones de los stakeholders que son Significativas para la arquitectura?				
8	¿Se seleccionan los puntos de vistas arquitectónicos?				
9	¿Cada punto de vista especifica: nombre del punto de vista, los stakeholders a ser tratados, las preocupaciones a ser tratadas, el lenguaje, técnicas de modelado o métodos analíticos empleados para construir una vista basada en el punto de vista, la fuente (podría incluir al autor, la fecha, la referencia a otros documentos; según determine la organización)?				
10	¿Si posee una o mas vista arquitectónica?				
11	¿Si cada vista en la descripción de la arquitectura se corresponde con un punto de vista?				
12	¿Cada vista en la descripción de la arquitectura se realizara conforme a las especificaciones del punto de vista correspondiente?				
13	¿Cada vista arquitectónica incluye información sobre el identificador y otra información introductoria definida por la organización?				
14	¿Cada vista arquitectónica incluye una representación del sistema construido con los métodos, lenguaje, técnicas analíticas y de modelado vinculados con la vista?				
15	¿Cada vista arquitectónica muestra información sobre				

	la configuración definida por la organización?				
16	¿Se lleva un registro de todas las inconsistencias encontradas entre los componentes de la descripción de la arquitectura?				
17	¿La descripción de la arquitectura contiene un registro con las consistencias de las vistas arquitectónicas de esa descripción de la arquitectura?				
18	¿Se realiza un análisis de las consistencias a través de las vistas de la arquitectura?				
19	¿La descripción de la arquitectura tiene incluido la justificación de los conceptos arquitectónicos seleccionados?				

Anexo 9 Lista de chequeo para la Revisión al Diseño en el Simulador Quirúrgico.

	Criterio de Evaluación	S	N	NP	Observación
1	¿Se realizó el diseño de las entidades?				
2	¿El lenguaje utilizado para especificar una clase de diseño es entendible?				
3	¿Están especificados todos los requerimientos en las entidades de diseño?				
4	¿Esta definido cualquier requisito en el diseño realmente?				
5	¿El diseño está correcto, consistente con los requisitos y muestra una adecuada trazabilidad hacia los mismos?				
6	¿Permite el modelo de diseño una comprensión en profundidad de los aspectos relacionados con los requisitos no funcionales?				
7	¿Se realizó la selección y descripción de los atributos?				

8	¿Se cuenta con alguna vista de diseño en el proyecto?				
9	Si existe alguna vista ¿que tipos de vista de diseño se utiliza?				
10	¿Las vistas de diseño cumplen con los requerimientos de alcance, uso y representación?				
11	¿El diseño implementa una secuencia correcta de eventos, entradas, salidas, interfaces, flujo lógico?				
12	¿El diseño es capaz de descomponer los trabajos de implementación en partes más manejables?				
13	¿Se cumplieron las actividades de Análisis y Diseño?				
14	¿Están documentados los artefactos de Análisis y diseño?				
15	¿Se realizó y documentó un diseño detallado de las interfaces?				
16	¿Un subsistema de diseño define las dependencias sobre otros subsistemas o interfaces de otros subsistemas?				
17	¿Está actualizada la documentación del usuario por parte del desarrollador?				
18	¿Un subsistema de diseño define las interfaces que han de ser proporcionadas por el subsistema?				
19	¿Se especifica las operaciones de las clases de diseño?				

Anexo 10 Lista de Chequeo para la Revisión a la Documentación.

	Criterio de Evaluación	S	N	NP	Observación
1	¿En el glosario de términos aparece la definición de los conceptos básicos que se deben dominar				

	relacionados con el proyecto?				
2	¿En la plantilla de especificaciones de requisitos se definieron los requerimientos funcionales y no funcionales del sistema?				
3	¿Se definieron las actividades a automatizar o Modelo de Dominio y concordancia de los requisitos?				
4	¿Existe concordancia de los requisitos con el alcance del proyecto definido en el Documento Visión?				
5	¿Existe concordancia de los requisitos con los casos de uso definidos en el Modelo de Casos de Uso del sistema?				
6	¿Hay conformidad con el esquema del expediente de proyecto?				
7	¿Están completos los documentos del expediente de proyecto según su avance?				
8	¿Existe un establecimiento y cumplimiento del cronograma del proyecto en función de la definición del plan de desarrollo?				
9	¿En el plan de desarrollo de software se identifica el Polo Productivo al que pertenece el proyecto y se define el tipo de proyecto y la metodología de desarrollo?				
10	¿Se establece el alcance del proyecto?				
11	¿Se encuentra identificada la lista de riesgos?				
12	¿Se establece la estructura organizacional, los roles y las responsabilidades de los miembros del proyecto?				
13	¿Existe una correspondencia del rol con las tareas asignadas?				
14	¿Hay vinculación con los procesos docentes e investigativos?				
15	¿Se emplea la metodología seleccionada y la notación UML?				

16	¿Se registra el capital humano, los recursos materiales y los resultados del proyecto?				
17	¿Se establece el plan de aseguramiento de la calidad?				
18	¿Existe un registro del tiempo de trabajo de los desarrolladores y de los defectos detectados?				
19	¿Se registran las estimaciones realizadas?				
20	¿Se establece la gestión de la configuración?				
21	¿En el documento de la arquitectura de software se definen las herramientas? ¿Estas herramientas se emplean?				
22	¿Se definen y se emplean las tecnologías y herramientas específicas en el documento de la arquitectura de software?				
23	¿En el documento de la arquitectura de software se especifican los requerimientos de hardware?				

Anexo 11 Lista de chequeo para la Revisión a la Gestión de Configuración en el Simulador Quirúrgico

	Criterio de Evaluación	S	N	NP	Observación
1	¿Se definen cuales son los tipos de elementos de configuración que se obtendrán en el sistema?				
2	¿Se realiza el control de versiones sobre los elementos de configuración y se definen herramientas para este fin?				
3	¿Se aplica un proceso de toma de decisiones sobre cualquier elemento de configuración?				
4	¿Se documentan todos los cambios realizados?				
5	¿Existe una persona que aprueba los cambios?				

	realizados?				
6	¿Se comunican los cambios al resto del equipo y esta definido un procedimiento para esto?				
7	¿Se documentan todos los elementos que son afectados con el cambio?				
8	¿Una vez implementado un cambio se aprueba nuevamente?				
9	¿Esta definida las líneas base del proyecto?				
10	¿Se controla las liberaciones parciales o totales del producto (release)?				
11	¿Se elaboro un Plan de Gestión de la Configuración desde la etapa inicial del proyecto?				
12	¿El plan tiene descrito las actividades, procedimientos y cronograma de gestión de la configuración?				
13	¿Se especifican los responsables de ejecutar estas actividades?				
14	¿Se identifican los artículos de software y las versiones que serán controladas por el proyecto?				
15	¿Para cada artículo de software y sus versiones se ha especificado las referencias de versión y la documentación que lo identifica?				
16	¿Se esta llevando a cabo el control de la configuración?				
17	¿Están identificadas y registradas las solicitudes de cambio?				
18	¿Se ejecuta el análisis y la evaluación de la solicitud de cambios?				
19	¿Se ejecuta y registra el proceso de aprobación o desaprobación de la solicitud?				
20	¿Se ejecuta y registra el proceso de verificación y				

	liberación del producto modificado?				
21	¿Se realiza la contabilidad del estado de la configuración?				
22	¿Se almacenan en un registro el estado y el historial de los artículos de software controlados?				
23	¿Los informes constan de la siguiente información: número de cambios al proyecto, últimas versiones del artículo de software, los identificadores de la liberación, el número de liberaciones y las comparaciones entre liberaciones?				
24	¿Se verifica la integridad funcional del artículo de software de acuerdo con los requisitos?				
25	¿Se realiza una apropiada gestión de la liberación y entrega?				
26	¿Se conservan las copias maestras del código y la documentación?				

Anexo 12 Registro de No Conformidades

1. Aspectos generales

[Descripción de Aspectos Generales a tener en cuenta a la hora de analizar el resultado de las pruebas, incidencias en el momento de su desarrollo y otros aspectos relevantes.]

1.1 Elementos Probados.

[Descripción general o lista de los Elementos Probados, y otros aspectos importantes a tener en cuenta a la hora de analizar las No Conformidades Detectadas.]

1.2 Elementos no Probados y causas.

[Descripción general o lista de los Elementos no Probados, la causa de que no se hayan podido realizar las Pruebas y cualquier otro elemento importante que aporte la información necesaria para que sean analizadas estas causas y resueltas para la siguiente iteración.]

2. Tabla de No Conformidades Detectadas

Elemento	No	No conformidad	Aspecto correspondiente	Etapas de detección	Importancia	Recomendación
<Nombre del Elemento>	< 1>	<Descripción de la No Conformidad>	<Descripción del Aspecto correspondiente>	<Etapas de detección del error>	<X>	<X>

3. Recomendaciones

Elemento	No	No conformidad	Aspecto correspondiente	Etapas de detección
<Nombre del Elemento>	< 1>	<Descripción de la No Conformidad>	<Descripción del Aspecto correspondiente>	<Etapas de la detección del error>

Anexo 13 Registro de Respuesta a las No Conformidades

1. Aspectos generales

1.1 Elementos Probados.

1.2 Elementos no Probados y causas.

2. Tabla de No Conformidades Detectadas

Elemento	No	No conformidad	Aspecto correspondiente	Etapa de detección Código del CF	Cumplimiento	Observaciones

3. Recomendaciones

Elemento	No	No conformidad	Aspecto correspondiente	Etapa de detección
<Nombre del Elemento>	< 1>	<Descripción de la No Conformidad>	<Descripción del Aspecto correspondiente>	<Etapa de la detección del error>

Anexo 14 Encuesta Realizada para la Validación

1. ¿En qué medida considera UD correcto el desarrollo de la estrategia propuesta?

Alta___ Media___ Baja___

2. ¿En qué nivel los roles y responsabilidades propuestos pueden lograr eficiencia y máxima calidad en el proyecto Simulador Quirúrgico?

Alta ___ Media ___ Baja ___

3. ¿En qué medida considera UD correcta las condiciones necesarias que se describen para que la investigación sea desarrollada con éxito?

Alta ____ Media ____ Baja ____

4. ¿En qué nivel considera UD correcta las actividades descritas en la estrategia?

Alta ____ Media ____ Baja ____

GLOSARIO DE TÉRMINOS Y SIGLAS

Acción correctiva: Acción para eliminar la causa de una no conformidad detectada u otra situación no deseada.

Artefacto: Pedazo de información producido, modificado o usado por un proceso. Son resultados tangibles del proyecto, las cosas que se van creando y usando hasta obtener el producto final.

Calidad: Capacidad de cumplir con las características inherentes de un producto, componente de proceso, o proceso para satisfacer las exigencias de los clientes.

Control de la calidad: actividad que se realiza para detectar los defectos del producto que se desarrolla.

Conformidad: Cumplimiento de un requisito.

CMA: Cirugía de Mínimo Acceso.

CMMI: Modelo de Madurez de Capacidad Integrado.

Flujo de trabajo: secuencia de actividades realizadas por trabajadores y que produce un resultado de valor observable.

IEEE: Instituto de Ingenieros Eléctricos y Electrónicos.

ISO: Organización Internacional para la Estandarización.

LMC: Lineamientos Mínimos de Calidad.

Metodología: Un sistema de principios y normas generales de organización y estructuración teórico-práctica de actividades.

Métricas: Continúa aplicación de técnicas basadas en la medición al proceso de desarrollo de software y a sus productos para proveer información administrativa significativa y oportuna.

No conformidad: Incumplimiento de un requisito.

Proceso: Conjunto de actividades mutuamente relacionadas o que interactúan, las cuales transforman elementos de entrada en resultados, consumiendo recursos.

Proyecto: Proceso único consistente en un conjunto de actividades coordinadas y controladas con fechas de inicio y de finalización, llevadas a cabo para lograr un objetivo conforme con requisitos específicos.

Registro: Documentación que proporciona evidencia objetiva de las actividades realizadas o de los resultados logrados.

Requisito: Condición o cualidad que debe cumplir alguien o algo

Revisión: Reuniones de un grupo definido de personas cuyo objetivo es encontrar errores en un artefacto de software.

RTF: Revisión Técnica Formal.

Rol: Define el comportamiento y responsabilidades de un individuo o un grupo de individuos que trabajan en equipo.

SQA: Aseguramiento de Calidad.

SVN: Subversion.

UCI: Universidad de las Ciencias Informáticas.

